May, 1969

OPTIMALITY PROPERTIES OF VARIOUS PROCEDURES FOR RANKING *n* DIFFERENT NUMBERS USING ONLY BINARY COMPARISONS

by

Abdollah Hadian

Technical Report No. 117

University of Minnesota
Minneapolis, Minnesota

Submitted as a Thesis to the Faculty of the Graduate School of the University of Minnesota in partial fulfillment of the requirement for degree of Doctor of Philosophy.

Research supported by National Science Foundation Grant GP-9018.

ACKNOWLEDGEMENTS

The author wishes to express his sincere gratitude to Professor Milton Sobel for his expert guidance and encouragement during the preparation of this dissertation, and whose desire for research is inspiring. I would also like to express my thanks to Professor Robert J. Buehler for reading the manuscript and making helpful suggestions.

TABLE OF CONTENTS

ABSTRACT (i)

CHAPTER I: General Results for the Problem of Ranking n Items

1.1 Statement of the problem 1
1.2 E-optimality . . . . . 1 – 4
1.3 M-optimality (minimax) . . . . . 4 – 8
1.4 Relations between the E-noiseless and M-noiseless properties 8 – 10
1.5 Subclasses of procedures . . . . . 10 – 12

CHAPTER II: On the Inductive Class $\mathcal{I}$

2.1 Introduction 13
2.2 The inductive procedure R_I and its expectation . . . . . 13 – 16
2.3 An explicit expression for $S(n)$ and its asymptotic behavior . . 16 – 19
2.4 On an inductive procedure $R_S^{(t)}$ . . . . . 19 – 20

CHAPTER III: On the Semi-inductive Procedure R_{FJ}

3.1 Introduction 21
3.2 An explicit expression for $U(n)$ . . . . . 21 – 23
3.3 A simpler explicit expression for $U(n)$ . . . . . 23 – 25
3.4 Lower bounds for all procedures in the class $\mathcal{I}$ . . . . . 25 – 28
3.5 Recursive relations for $F(n) \equiv A(n|R_{FJ})$ . . . . . 28 – 32

APPENDIX 33 – 34

REFERENCES 35 – 36

ABSTRACT

OPTIMALITY PROPERTIES OF VARIOUS PROCEDURES FOR RANKING n DIFFERENT NUMBERS USING ONLY BINARY COMPARISONS

There are $n \geq 2$ unknown real numbers which are pairwise unequal. Starting from a random order, we want to rank the t largest ($1 \leq t \leq n-1$) of these numbers by using only binary errorless comparisons. After each comparison, the experimenter is only told which is larger (or equivalently, smaller). The problem is to obtain a procedure for finding and ordering the t largest numbers which is such that the (random) number of comparisons is small in some well-defined sense.

The main part of this work is concerned with the case $t = n - 1$. Two optimality criteria are introduced: to minimize the expected number of comparisons and to minimize the maximum number of comparisons. It is found that

$$H(n!) = \lceil \log(n!) \rceil - \frac{2^{\lceil \log(n!) \rceil} - n!}{n!}$$

and $\lceil H(n!) \rceil$ are lower bounds for the first and second criteria, respectively, under any procedure (here all logs are to base 2 and $\lceil x \rceil$ is the smallest integer greater than or equal to x). All the procedures whose expectation achieves the lower bound $H(n!)$ are called E-noiseless and are characterized in Theorem 1a by the fact that the range (or difference between the maximum and minimum number of comparisons) is exactly one. It is proved in Theorem 2a that the maximum number of comparisons of E-noiseless procedures achieves the lower bound $\lceil H(n!) \rceil$, but the converse need not be true.

In Chapter 2 a certain subclass $\mathcal{I}$ of inductive procedures is introduced and all the optimal procedures in $\mathcal{I}$ are found (Theorems 3 and 4). In particular, it is noted that the Steinhaus procedure R_S is one of the optimal procedures in the subclass $\mathcal{I}$.

In Chapter 3 a procedure R_{FJ} due to Ford and Johnson (see reference 5) is studied. An explicit expression for the maximum number of comparisons under R_{FJ} is found and it is shown that this maximum is strictly smaller than the maximum under R_S for $n \geq 5$. A subclass of procedures $\mathcal{J}$, which includes R_{FJ}, is considered and lower bounds for all procedures in $\mathcal{J}$ are found. For $n = 7$ an optimal procedure in the subclass $\mathcal{J}$ is given which proves that R_{FJ} does not have minimum expectation in the subclass $\mathcal{J}$, let alone among all procedures.

Milton Sobel

CHAPTER I
General Results for the Problem of Ranking n Items

1.1. Statement of the problem

There are n (an integer greater than or equal to 2) unknown real numbers $x_1, x_2, \ldots, x_n$, each called an item, which are pairwise unequal. Starting from a random order we want to rank the t largest ($1 \leq t \leq n-1$) of these items by means of a sequence of binary comparisons. After each comparison the experimenter is told merely which of the two items compared is larger (or equivalently smaller). The problem is to find a procedure R for ranking (i.e., finding and ordering) the t largest items in such a way that the required number of comparisons should be small in some well-defined sense.

The items $x_1, x_2, \ldots, x_n$ can be regarded as different comparable quantities such as weights or abilities of n players in a tournament in which each game involves two opposing players (such as tennis). In the first instance comparisons are made by means of a two-pan scale, allowing only one object in each pan; in the second instance the comparisons are made by having two players play against each other. We assume that the better player always wins; so that no game can end in a draw and transitivity is used whenever possible. We also assume that our procedures will not include any comparison for which the result is known or can be inferred from known results.

Special cases of this problem are:

(a) $t = 1$, which is relatively trivial and in the literature is mostly referred to as a knock-out tournament (see, for example, [4]).

(b) $t = 2$, which is studied by M. Sobel [16].

(c) $t = n - 1$, i.e., ranking all items using only binary comparisons.

The main part of this work is concerned with the case $t = n - 1$; nevertheless the concepts and notations are introduced for the case of general t.

The number of comparisons, under any procedure, required to rank the t largest of n items is a discrete random variable (possibly degenerate in some cases). For example, if $n = 3$ and $t = 2$ any procedure requires 2 or 3 comparisons with probability 1/3 or 2/3, respectively. Thus we may consider two optimality properties described in the following sections.

1.2. E-optimality

Let $A_t(n|R)$ be the expected number of comparisons required to rank the t largest of n items under a procedure R. Let

$$L(t,n) = \min_{R \in \mathscr{R}_t} A_t(n|R) \tag{1.1}$$

where R is a procedure in the (finite) class $\mathscr{R}_t$ of all possible procedures. For $t = n - 1$ we denote $A_{n-1}(n|R)$ by $A(n|R)$, $\mathscr{R}_{n-1}$ by $\mathscr{R}$, and $L(n-1,n)$ by $L(n)$.

A procedure R is called *E-optimal* if for each n and t,

$$A_t(n|R) = L(t,n). \tag{1.2}$$

Since the result of any comparison of two items is the knowledge of one being greater or smaller than the other, a procedure can be represented as a tree (or a directed graph) without any loops and having exactly two forks at each node; we refer to this simply as a tree. $A_t(n|R)$ then corresponds to the average (branch) length of the tree that represents a procedure. For example, a procedure for ranking 3 items ($n = 3, t = 2$) is given by the tree in Fig. 1. For any of our trees, the arrow pointed to the left under x_i vs. x_j (to be read 'x_i versus x_j') indicates the result that $x_i < x_j$. Similarly, the arrow pointed to the right under x_i vs. x_j indicates the result that $x_i > x_j$.

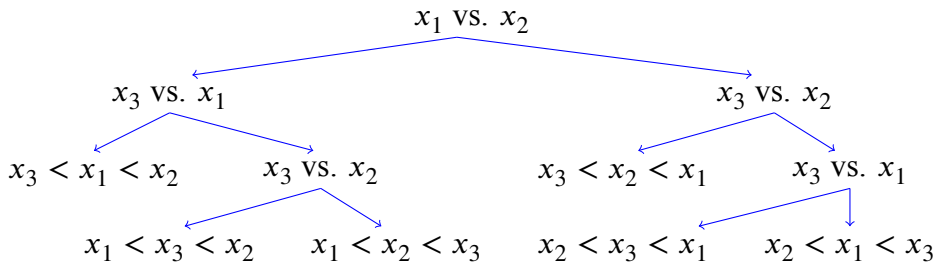


Fig. 1. A procedure for ranking 3 items; x_1, x_2, and x_3.

The end points of this tree correspond to the $3! = 6$ permutations of x_1, x_2, and x_3. For the procedure R_0 given in Fig. 1 we have $A(3|R_0) = 2 \cdot \frac{2}{6} + 3 \cdot \frac{4}{6} = \frac{8}{3}$.

Procedures that have the same expected number of comparisons are called *E-equivalent*. The *E-excess* of a procedure R, denoted by $C_t(n|R)$, is defined to be

$$C_t(n|R) = A_t(n|R) - L(t,n) \tag{1.3}$$

where $L(t,n)$ is defined by (1.1). $L(t,n)$ in general is unknown. Special cases for which $L(t,n)$ is known or conjectured to be known are the following:

(a) $t = 1$, where $L(1,n) = n - 1$ (see, for example, [4] and [17]).

(b) $t = 2$, where $L(2,n)$ is conjectured to be $n + r - 2$ for $n = 2^r$ (see [16]).

(c) $t = n - 1$, in this case

$$L(n) = H(n!) \text{ for } n \le 6, n = 9, 10, \tag{1.4}$$

where for any integer $m \ge 1$,

$$H(m) = \lfloor \log(m) \rfloor + 2\left(\frac{m - 2^{\lfloor \log(m) \rfloor}}{m}\right) = \lceil \log(m) \rceil - \frac{2^{\lceil \log(m) \rceil} - m}{m}. \tag{1.5}$$

Here $\lfloor x \rfloor$ denotes the largest integer not greater than x, and $\lceil x \rceil$ denotes the smallest integer not less than x. All logarithms used in this work are to base 2 unless otherwise stated. Picard [11] has found the values of $L(n)$ for $n \le 6$, and Y. Cesari (private communication) has found $L(n)$ for $n = 9$, 10, both by exhibiting trees whose expected length is equal to $H(n!)$ for $n \le 6$ and $n = 9$, 10. $H(m)$ given in (1.5) is called the Huffman lower bound (see [11], [14], [15], and [16]). Other equivalent forms of $H(m)$ which will be used in our later discussions are

$$\begin{aligned} H(m) &= r + \frac{2c}{m}, \\ H(m) &= s - \frac{d}{m}, \end{aligned} \tag{1.6}$$

where non-negative integers $r = r_m$, $c = c_m$, $s = s_m$, and $d = d_m$ are defined by

$$(1.7)\quad \begin{aligned} m &= 2^r + c \quad (0 \le c < 2^r), \\ m &= 2^s - d \quad (0 \le d < 2^{r-1}). \end{aligned}$$

The E-optimality of any procedure with expected number of comparisons equal to $H(n!)$ is proved in Lemma 1 below. We mention that the procedure given by Steinhaus in [18], which is identical with the R_{FJ} procedure in [5] for $n = 5$, is E-optimal even though it was introduced to have a different optimality (see next section) property.

The *E-noise* $B(n|R)$ of a procedure R is defined by

$$(1.8)\quad B(n|R) = A(n|R) = H(n!).$$

The procedure R is called *E-noiseless* if $B(n|R) = 0$. It should be noted that an E-noiseless procedure may not exist whereas an E-optimal procedure always exists.

Lemma 1. If a procedure is E-noiseless then it is E-optimal.

Proof. First we show that

$$(1.9)\quad H(n!) \le L(n) \le A(n|R).$$

In fact, for any problem with m states of nature, one of which is true, $H(m)$ represents the minimum expected number of ('yes or no') questions needed to find the true one when there is no restriction whatsoever on the question that can be asked, i.e., we can choose any subset and ask if the true state is contained in it (see [11], [12], and [14]). On the other hand in our problem of ranking n items ($t = n - 1$) we have $m = n!$ states of nature and we can only ask questions (or make partitions) that correspond to some binary comparison. This already proves $H(n!) \le A(n|R)$ and $H(n!) \le L(n)$. Hence (1.9) follows easily.

Now consider any procedure R for our problem, which is E-noiseless. Then

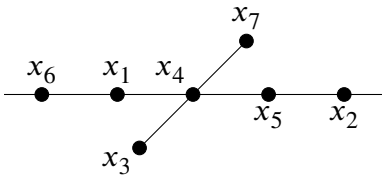
$$(1.10)\quad 0 = A(n|R) - H(n!) \ge A(n|R) - L(n) \ge 0,$$

and hence $A(n|R) = L(n)$, i.e., R is E-optimal.

To briefly illustrate the fact that certain partitions do not correspond to a comparison we assume that we wish to rank 7 items $x_1, x_2, \ldots, x_7$ which are known to satisfy

$$(1.11)\quad x_6 < x_1 < x_4 < x_5 < x_2 \text{ and } x_3 < x_4 < x_7.$$

These inequalities in (1.11) can also be shown by the following diagram:



Here there are $3 \cdot 3 = 9$ states of nature which are all possible permutations of 7 items, consistent with (1.11). The next comparison has to be x_3 vs. x_6 or x_3 vs. x_1 or x_7 vs. x_5 or x_7 vs. x_2,

and these partition the 9 possible states of nature into subgroups of sizes $(3,6)$, $(6,3)$, $(3,6)$, and $(6,3)$, respectively. In particular we note that there is no comparison which corresponds to the 'crucial' partition $(4,5)$, nor to any of the other partitions such as $(8,1)$, $(7,2)$, etc. Suppose we want to rank 7 items satisfying the inequalities in (1.11). It can be easily checked that in this case any procedure is as good as the others and requires, on average, $2\left(1\cdot\frac{1}{3}+2\cdot\frac{2}{3}\right)=3+\frac{1}{3}$ comparisons. Whereas if the partition $(4,5)$ had been allowed we would have needed, on average, $1+\frac{4}{9}H(4)+\frac{5}{9}H(5)=3+\frac{2}{9}=H(9)$ questions.

The above lemma and illustration indicate that a procedure may simultaneously have positive E-noise and be E-optimal. In fact we conjecture that the procedure R_1 given below for $n=7$ (starting from a random order) is such a procedure. The E-noise for R_1 is given by

$$B(7|R_1)=12+\tfrac{121}{315}-H(7!)=\tfrac{1}{105}=.0095. \tag{1.12}$$

Cesari in [2] has proved that it is impossible to have a procedure for ranking 7 items and which has zero E-noise. He has given a procedure equivalent to R_1 (i.e., with the same amount of noise but not identical to it).

For procedure R_1 we start with the following comparisons: x_1 vs. x_2; x_3 vs. x_4; and x_5 vs. x_6, assuming, without loss of generality, that $x_1<x_2$, $x_3<x_4$, and $x_5<x_6$. Then we compare x_2 vs. x_4 and assume $x_2<x_4$. The continuation of the procedure R_1 after these 4 comparisons is represented by the tree in Fig. 2 (further explanations of the numbers shown are given in the appendix).

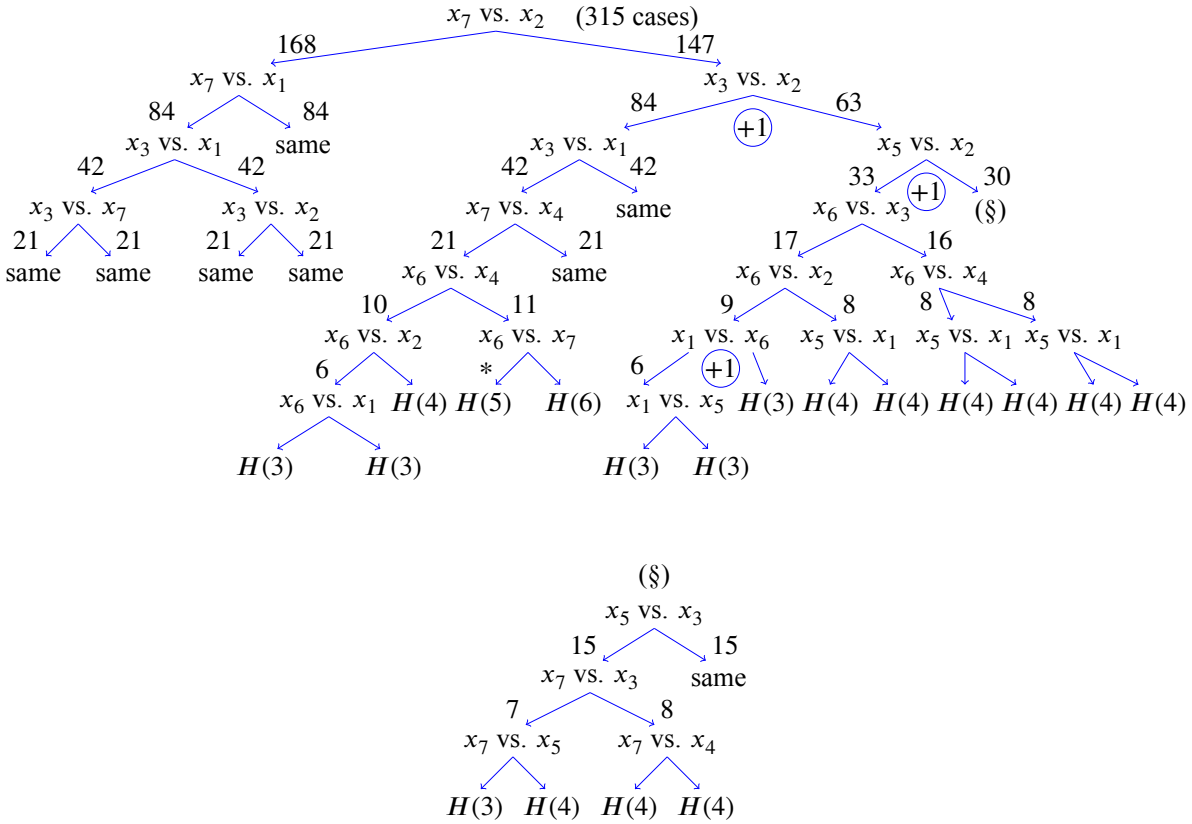


Fig. 2. A tree for the continuation of procedure R_1 for ranking 7 items; conjectured to be E-optimal.

1.3. M-optimality (minimax)

Let $M_t(n|R)$ be the maximum number of comparisons required to find and order the t largest of n items under a procedure R. Let

$$(1.13)\quad L_{\text{Max}}(t,n) = \min_{R\in\mathscr{R}_t} M_t(n|R)$$

where R is a procedure belonging to the (finite) class $\mathscr{R}_t$ of all possible procedures. For $t = n-1$ we denote $M_{n-1}(n|R)$ by $M(n|R)$, $\mathscr{R}_{n-1}$ by $\mathscr{R}$ and $L_{\text{Max}}(n-1,n)$ by $L_{\text{Max}}(n)$.

A procedure R is called *M-optimal* if for each n and t,

$$(1.14)\quad M_t(n|R) = L_{\text{Max}}(t,n).$$

All procedures having the same maximum number of comparisons are called *M-equivalent*. When the complete procedure R is written as a tree, $M_t(n|R)$ corresponds to the maximum (branch) length of this tree.

The *M-excess* of a procedure R, denoted by $D_t(n|R)$, is defined to be

$$(1.14a)\quad D_t(n|R) = M_t(n|R) - L_{\text{Max}}(t,n).$$

In general $L_{\text{Max}}(t,n)$ is unknown; special cases in which $L_{\text{Max}}(t,n)$ is known are the following:

(a) $t = 1$, where $L_{\text{Max}}(1,n) = L(1,n) = n-1$.

(b) $t = 2$, where $L_{\text{Max}}(2,n) = n - 2 + \lceil \log(n) \rceil$ (see [13], [16], and reference 20 in [16]). We notice from (b) in Section 1.2 that

$$(1.15)\quad L_{\text{Max}}(2,n) = L(2,n) = n + r - 2 \text{ for } n = 2^r.$$

M. Sobel has introduced several procedures for this case, two of which achieve $L_{\text{Max}}(2,n)$ and hence are M-optimal.

(c) $t = n-1$ for $n \le 11$ and $n = 20, 21$. It is shown by Ford and Johnson in [5] that

$$(1.16)\quad L_{\text{Max}}(n) = \lceil \log(n!) \rceil \text{ for } n \le 11 \text{ and } n = 20, 21.$$

They have found $L_{\text{Max}}(n)$ by introducing a specific procedure, R_{FJ}, for all n, which attains the lower bound in (1.16), i.e.,

$$(1.17)\quad M(n|R_{FJ}) = \lceil \log(n!) \rceil \text{ for } n \le 11 \text{ and } n = 20, 21.$$

The fact that $\lceil \log(n!) \rceil$ is a lower bound for the maximum number of comparisons under any procedure is proved in [5] and Theorem 24 of [9a], and proved at length in Theorem 1 of [9]. We now give another proof in the following lemma based on (1.9).

Lemma 2. For any procedure R we have

$$(1.18)\quad \lceil \log(n!) \rceil = \lceil H(n!) \rceil \le L_{\text{Max}}(n) \le M(n|R).$$

Proof. From (1.9) we have, for any procedure R,

$$(1.19)\quad H(n!) \le A(n|R) \le M(n|R).$$

In particular, letting R be the M-optimal procedure we have

(1.20) $H(n!) \leq L_{\text{Max}}(n)$.

Since $M(n|R)$ and $L_{\text{Max}}(n)$ are integers and since $L_{\text{Max}}(n) \leq M(n|R)$, using (1.19) and (1.20) we have

(1.21) $\lceil H(n!) \rceil \leq L_{\text{Max}}(n) \leq M(n|R)$.

To finish the proof of the lemma we note that for any integer m with $2^{s-1} < m \leq 2^s$ we have $H(m) = s - \theta$ where $0 \leq \theta < 1$. It follows that $\lceil H(m) \rceil = s = \lceil \log(m) \rceil$.

The *M-noise* $B_{\text{Max}}(n|R)$ of a procedure R is defined by

(1.22) $B_{\text{Max}}(n|R) = M(n|R) - \lceil H(n!) \rceil$.

The procedure R is said to be *M-noiseless* if its M-noise is zero. It follows from (1.21) that an M-noiseless procedure is M-optimal, but the converse does not necessarily hold. In fact, the case $n = 12$ is an example of this since we note that $\lceil H(12!) \rceil = 29$ and it has been shown by M. B. Wells (private communication) that $L_{\text{Max}}(12) \geq 30$. Since $M(n|R_{FJ}) = 30$, it follows that R_{FJ} is M-optimal even though it has one unit of M-noise.

To investigate the properties of M-noiseless procedures we need to solve an auxiliary problem which is of some interest *per se*.

Auxiliary Problem: There are $N \geq 2$ objects or possible states of nature. Each of these has probability $\frac{1}{N}$ of being the true state of nature which is unknown to us. We are to identify the true state of nature by a sequence of questions. We assume that after i questions ($i = 0, 1, 2, \ldots$) the true state of nature is among N_i states, where $N = N_0 > N_1 > N_2 > \ldots > 1$. If $N_i = 1$ for any i the true state of nature is identified; otherwise we partition the N_i states arbitrarily into two disjoint sets and ask in which of these two sets the true state of nature lies. The answer is given correctly. The problem is to find a procedure $T_N^{(M)}$ that minimizes the maximum number of questions. It is also of interest to find $G(N)$, the maximum number of questions asked under the procedure $T_N^{(M)}$.

Remark 1. A closely related problem would arise if we are interested in a procedure which minimizes the *expected* number of questions. This is a solved problem and is treated in [11], [12], [14], and [16]; we used this result in the proof of Lemma 1 above. In fact, let $T_m^{(E)}$ be a procedure that minimizes the expected number of questions given that the true state is among m objects ($2 \leq m \leq N$). Then, as in [16] under $T_m^{(E)}$, we select a set of size $y = y(m)$ for the next questions where y is any integer such that there is no power of 2 between y and $m - y$, i.e., if

(1.23) $\Delta_m^{(E)} = \{j \mid \text{there is no power of 2 strictly between the integers } j \text{ and } m - j\}$

then we select any y in $\Delta_m^{(E)}$. It is easy to see that $\Delta_m^{(E)}$ is never empty and consists of a single integer if and only if m is a power of 2. The expected number of questions under $T_m^{(E)}$ is $H(m)$, given by (1.5).

Remark 2. The auxiliary problem and also the expectation problem in Remark 1 are relatively easy problems because after each question we are left with a situation entirely similar to the original one. It should be noted that there are no restrictions on the size or nature of the set that we can choose

for the question. This is unlike the problem stated in Section 1.1, say for $t = n - 1$, where the only partitions (or questions) allowed are those that correspond to some comparison. There may exist 'crucial' partitions (or questions) that do not correspond to any comparison and consequently are not possible, as was shown in the paragraph following (1.10).

At an arbitrary point in the auxiliary problem we suppose that the true state of nature is among m states, where $2 \le m \le N$. In this unrestricted set-up a procedure is determined by knowing $y \equiv y(m)$, the size of the set to be chosen for the next question. Hence the procedure $T_m^{(M)}$ and also $G(m)$ can be found by the recursive formula

$$G(m) = 1 + \min_{1 \le y \le m-1} \max(G(y), G(m-y)) \tag{1.24}$$

with the boundary condition

$$G(1) = 0. \tag{1.25}$$

Let $\Delta_m^{(M)}$ be the closed interval

$$\Delta_m^{(M)} = \left[m - 2^{\lceil \log(m/2) \rceil}, 2^{\lceil \log(m/2) \rceil}\right], \tag{1.26}$$

which is never void and reduces to a single integer if and only if m is a power of 2.

Lemma 3. The solution of (1.24) and (1.25) is given by

(a) $G(m) = s_m = \lceil \log(m) \rceil$ $(m = 1, 2, \ldots)$.

(b) An integer y minimizes the RHS of (1.24) if and only if y belongs to $\Delta_m^{(M)}$, defined in (1.26).

Proof. Trivially $\lceil \log(m) \rceil$ satisfies (1.25). To prove the rest of the lemma we consider the case $y \le m - y$ or $y \le \frac{m}{2}$; later we interchange y and $m - y$, if necessary, to get the results for the case $y \ge m - y$. Since $y \le m - y$,

$$\max(\lceil \log(y) \rceil, \lceil \log(m-y) \rceil) = \lceil \log(m-y) \rceil$$

and

$$\min_{1 \le y \le m/2} \lceil \log(m-y) \rceil = \left\lceil \log\left(\left\lceil \frac{m}{2} \right\rceil \right) \right\rceil = \left\lceil \log\left(\frac{m}{2} \right) \right\rceil = \lceil \log(m) \rceil - 1. \tag{1.27}$$

It follows that (1.24) is satisfied for $1 \le y \le \frac{m}{2}$. A similar argument finishes the proof of (a).

To prove (b) for this case it suffices to find all integers y with $1 \le y \le \frac{m}{2}$ such that

$$\lceil \log(m-y) \rceil = \left\lceil \log\left(\frac{m}{2} \right) \right\rceil. \tag{1.28}$$

It is easily verified that (1.28) is satisfied if and only if

$$m - 2^{\lceil \log(m/2) \rceil} \le y \le \frac{m}{2}. \tag{1.29}$$

Changing y to $m - y$ for the alternative case we find that

$$\frac{m}{2} \le y \le 2^{\lceil \log(m/2) \rceil}. \tag{1.30}$$

Inequalities (1.29) and (1.30) finish the proof of (b). It should be noted that $\Delta_m^{(M)}$ always contains the closest integer (or integers) to $\frac{m}{2}$.

1.4. Relation between the E-noiseless and M-noiseless properties

A procedure may be M-noiseless and have E-noise greater than 0. For example, the R_{FJ} procedure in [5] for $n = 6$ is M-noiseless since its maximum number of comparisons is $\lceil H(6!)\rceil = 10$ and has E-noise equal to $A(6|R_{FJ}) - H(6!) = \left(19 + \frac{3}{5}\right) - \left(19 + \frac{26}{45}\right) = \frac{1}{45} > 0$ (see Table 2 in Section 3.4 below). For $n = 6$ Picard has given a procedure in page 116 of [11] which is both M-noiseless and E-noiseless.

The following theorem characterizes procedures that are both E-noiseless and M-noiseless. Let $\mu(n|R)$ be the minimum number of comparisons needed for ranking n items under a procedure R. The *range* $\rho(n|R)$ of a procedure is defined to be

(1.31) $\rho(n|R) = M(n|R) - \mu(n|R)$.

It is shown in Theorem 2 of [19] that for $n \geq 2$,

(1.32) $\rho(n|R) \geq 1$

for any procedure R that ranks n items.

Theorem 1. A procedure R for ranking n items is both E-noiseless and M-noiseless if and only if it has minimum range (i.e., if and only if $\rho(n|R) = 1$).

Proof. Let a_j denote the number of branches of length $k - j$ in a tree representing a procedure R, where $k = M(n|R)$. It is easily seen that we need at least $n-1$ comparisons so that $0 < a_0 < n!$, $a_j \geq 0$ for $j \geq 1$, and $a_j = 0$ for $j > k - n + 1$. Steinhaus also points out in [19] that the total number of branches is

(1.33) $\sum_{j=0}^{k-n+1} a_j = n!$,

and (as can be seen by increasing every branch to length k) we also have

(1.34) $\sum_{j=0}^{k-n+1} 2^j a_j = 2^k$.

Also, by definition the expected length is

(1.35) $A(n|R) = \frac{1}{n!}\sum_{j=0}^{k-n+1}(k-j)a_j = k - \frac{1}{n!}\sum_{j=1}^{k-n+1} j a_j$.

Finding a_1 from (1.33) and (1.34) and substituting it in (1.35) we get

(1.36) $A(n|R) = k - \frac{2^k - n!}{n!} + \frac{1}{n!}\sum_{j=2}^{k-n+1}(2^j - j - 1)a_j$.

If the procedure R is M-noiseless then $k = \lceil \log(n!)\rceil$ and by (1.5) we have $k - \frac{2^k - n!}{n!} = H(n!)$. Since R is also E-noiseless then $A(n|R) = H(n!)$ and hence from (1.36) we get

(1.37) $\sum_{j=2}^{k-n+1}(2^j - j - 1)a_j = 0$.

Since the coefficients are non-negative it follows that $a_j = 0$ for $j \geq 2$, i.e., the range of R is one.

Conversely, if $a_j = 0$ for $j \geq 2$ then from (1.36) and (1.9) we have

$$(1.38)\quad H(n!) \leq A(n|R) = k - \frac{2^k - n!}{n!}.$$

Since by (1.18) $k \geq \lceil \log(n!) \rceil$ which equals, say, s, let $k = s + i$ where i is a non-negative integer. To prove that any procedure R satisfying (1.38) is both M-noiseless and E-noiseless, it suffices to show that $i = 0$ and equality holds in (1.38). To see this we need only to show that

$$(1.39)\quad s + i - \frac{2^{s+i} - n!}{n!} \leq s - \frac{2^s - n!}{n!} = H(n!),$$

and that equality holds if and only if $i = 0$. The inequality in (1.39) is equivalent to

$$(1.40)\quad n!i \leq 2^s(2^i - 1).$$

Since $i \leq 2^i - 1$ for all i and $n! \leq 2^s$ for all n the inequality (1.40) holds. Clearly equality holds in (1.40) if and only if $i = 0$, and this proves our theorem.

To prove a relation between E-noiseless and M-noiseless procedures we need an additional lemma, stated below.

Lemma 4. For each m ($m = 2, 3, \ldots$),

$$(1.41)\quad \Delta_m^{(E)} \subset \Delta_m^{(M)}$$

where $\Delta_m^{(E)}$ and $\Delta_m^{(M)}$ are given by (1.23) and (1.26), respectively.

Proof. Let $y \in \Delta_m^{(E)}$. Clearly

$$(1.42)\quad \min(y, m - y) \leq \frac{m}{2} \leq \max(y, m - y).$$

On the other hand, $2^{\lceil \log(m/2) \rceil - 1}$ and $2^{\lceil \log(m/2) \rceil}$ are the largest and smallest powers of 2, respectively, such that

$$(1.43)\quad 2^{\lceil \log(m/2) \rceil - 1} \leq \frac{m}{2} \leq 2^{\lceil \log(m/2) \rceil}.$$

To satisfy the condition that there is no power of 2 between y and $m - y$, it is necessary and sufficient that

$$(1.44)\quad 2^{\lceil \log(m/2) \rceil - 1} \leq \min(y, m - y) \leq \max(y, m - y) \leq 2^{\lceil \log(m/2) \rceil}.$$

The inequalities in (1.44) imply in particular that

$$(1.45)\quad m - 2^{\lceil \log(m/2) \rceil} \leq y \leq 2^{\lceil \log(m/2) \rceil},$$

i.e., that $y \in \Delta_m^{(M)}$, and this proves our lemma.

Theorem 2. If a procedure is E-noiseless it is also M-noiseless.

Proof. Consider any non-terminal point reached in the course of carrying out the comparisons and suppose that we now have m possible states of nature (or cases), one of which is the true ordering of the n items. At the outset, $m = n!$. Let $y = y(m; x_i, x_j)$ be the size of the subset that contains the true ordering as a result of the comparison x_i vs. x_j. Using Lemma 2 in Section 6 of

[16] we conclude that a procedure is E-noiseless if and only if $y \in \Delta_m^{(E)}$. Similarly from Lemma 3 we conclude that a procedure is M-noiseless if and only if $y \in \Delta_m^{(M)}$. Hence Lemma 4 completes the proof of our theorem.

Remark 1. Note that the converse of Theorem 2, which is one of the questions raised in page 47 of [10], is not necessarily true. The Ford and Johnson procedure in [5] for $n = 6$, given in the beginning of this section, provides a counter-example.

Remark 2. Theorem 2 can be used in searching for an E-noiseless procedure, since it shows that we can restrict our attention to M-noiseless procedures.

Now we can combine Theorems 1 and 2 to strengthen them as follows:

Theorem 1a. A procedure for ranking n items is E-noiseless if and only if it has minimum range (i.e., if and only if the procedure is of range 1).

Theorem 2a. If a procedure is E-noiseless it is M-noiseless, and only M-noiseless procedures of range 1 are E-noiseless.

1.5. Subclasses of procedures

In previous sections we introduced two different optimality criteria and mentioned that existing procedures are 'optimal' only for certain values of n. The idea behind introducing subclasses is that we may be able to find an M-optimal (or E-optimal) procedure in a subclass when it is difficult to find an M-optimal (or E-optimal) procedure in $\mathscr{R}_t$. This idea turns out to be helpful since we may be able to find lower bounds for the subclass when lower bounds for the whole class are difficult to obtain. We may also be able to use a systematic method to find an 'optimal' procedure for the subclass whereas it seems to be a difficult task to invent a method for finding the 'optimal' procedure in the unrestricted class $\mathscr{R}_t$. First we introduce some terminology and definitions.

We recall that if $n = 2^r$ then a knock-out tournament finds the largest item among $x_1, x_2, \ldots, x_n$ by making $2^r - 1$ comparisons as follows: 2^{r-1} comparisons in the first round as x_1 vs. $x_2, \ldots, x_{n-1}$ vs. x_n; 2^{r-2} comparisons in the second round pairing off the winners (larger items) of the first round and so on until the winner is found. Without loss of any generality it is possible to assume that the winners of the first round are those with even indices, the winners of the second round are those with indices that are multiples of 4, etc. This notation is convenient.

Complete pairing: Let $n = 2^{r_1} + 2^{r_2} + \ldots + 2^{r_\alpha}$ be the binary expansion of n, i.e., 2^{r_1} is the highest power of 2 in n and 2^{r_j} is the highest power of 2 in $n - 2^{r_1} - \ldots - 2^{r_{j-1}}$, where $2 \le j \le \alpha$. The complete pairing consists of $\sum_{j=1}^{\alpha}(2^{r_j} - 1)$ comparisons, consisting of a knock-out tournament for each subset of size 2^{r_j} $(j = 1, 2, \ldots, \alpha)$. Without loss of generality we can take $x_1, x_2, \ldots, x_{\beta_1}$ for the first 'group', where $\beta_1 = 2^{r_1}$; and for $j \ge 2$ we can take $x_{i+1}, x_{i+2}, \ldots, x_{i+\beta_j}$ for the j^{th} 'group', where $i = i(j) = 2^{r_1} + \ldots + 2^{r_{j-1}}$ and $\beta_j = 2^{r_j}$. After the complete pairing of n items we are left with α configurations; for example, for $n = 13 = 2^3 + 2^2 + 2^0$ we have the 3 configurations given in Fig. 3.

Fig. 3. Complete pairing of 13 items.

The number of comparisons for complete pairing of n items $p = p(n)$, as given by the lemma in Section 4 of [16], is

(1.46) $p = n - \alpha = \sum_{j=1}^{\lfloor \log(n) \rfloor} \left\lfloor \frac{n}{2^j} \right\rfloor = \pi(n!)$

where α is the number of 1's in the binary expansion of n and $\pi = \pi(i)$ is the largest integer such that 2^π divides i, i.e., π and $c = c(i)$ are non-negative integers defined by

(1.47) $i = 2^\pi(2c + 1)$.

Ordinary pairing of n items consists simply of $\left\lfloor \frac{n}{2} \right\rfloor$ comparisons: x_1 vs. x_2, x_3 vs. x_4, ….

One may be interested in an E-optimal (M-optimal) procedure in a subclass $\mathscr{e}_t$ of $\mathscr{R}_t$ for ranking the t largest among n items. Let

(1.48) $L(t, n|\mathscr{e}_t) = \min_{C \in \mathscr{e}_t} A_t(n|C)$

and

(1.49) $L_{\text{Max}}(t, n|\mathscr{e}_t) = \min_{C \in \mathscr{e}_t} M_t(n|C)$.

Obviously,

(1.50) $L(t, n) \leq L(t, n|\mathscr{e}_t)$

and

(1.51) $L_{\text{Max}}(t, n) \leq L_{\text{Max}}(t, n|\mathscr{e}_t)$;

in particular, if $\mathscr{e}_t = \mathscr{R}_t$ we have equality in (1.50) and in (1.51). It is interesting to note that we may have equality in (1.50) (in (1.51)), which would imply that to find an E-optimal (M-optimal) procedure among all procedures one need only find an E-optimal (M-optimal) procedure in class $\mathscr{e}_t$ ('optimality' in a class $\mathscr{e}_t$ is defined in a similar fashion as 'optimality' in $\mathscr{R}_t$, i.e., the RHS's of (1.2) and (1.14) are replaced by $L(t, n|\mathscr{e}_t)$ and $L_{\text{Max}}(t, n|\mathscr{e}_t)$ respectively). More precisely, a class $\mathscr{C}_t$ is called *E-complete* (*M-complete*) if

(1.52) $L(t, n|\mathscr{C}_t) = L(t, n) \quad (L_{\text{Max}}(t, n|\mathscr{C}_t) = L_{\text{Max}}(t, n))$.

The class $\mathscr{R}_t$ is trivially both E- and M-complete.

Important subclasses that we consider are

- the inductive class $\mathscr{I}_t$: This class consists of all procedures that require, for fixed t and each n ($n = 2, 3, \ldots$), ranking of the t largest of $n - 1$ items inductively. For the special case $t = n - 1$ we denote $\mathscr{I}_{n-1}$ by $\mathscr{I}$. The class $\mathscr{I}$ is studied in Chapter 2.

- the semi-inductive class $\mathscr{J}_t$: This class consists of all procedures such as $R_{n,t}$ that require, for fixed t and each n ($n = 2, 3, \ldots$), the following 2 steps:

1. Ordinary pairing of n items, which as a result there will be $\left\lfloor \frac{n}{2} \right\rfloor$ larger items.
2. Ranking the t largest of $\left\lfloor \frac{n}{2} \right\rfloor$ larger items in step 1 according to $R_{\lfloor n/2 \rfloor, t}$.

For the special case $t = n - 1$ we denote $\mathscr{J}_{n-1}$ by $\mathscr{J}$. An example of procedures in $\mathscr{J}$ is the procedure given in [5]. We study the class $\mathscr{J}$ in Chapter 3.

- the complete pairing class $\mathscr{P}_t$: This class consists of all procedures that initially carry out the complete pairing on the n items. It is a proper subclass of $\mathscr{R}_t$.

  It is easy to show that $\mathscr{P}_t$ is an E- and M-sufficient class for $t = 1$. Let $\mathscr{P}_{n-1} \equiv \mathscr{P}$. We conjecture that $\mathscr{P}$ is an E- and M-complete class for ranking n items. We prove a somewhat weaker result in Lemma 5 below.

We may be interested in finding a procedure for ranking n items given a partial pre-existing order among n items. Suppose there are m permutations of n items consistent with the pre-existing order. A procedure R for ranking n items is said to be *conditionally E-noiseless* (*conditionally M-noiseless*) given a pre-existing order among items if the expected number of comparisons (maximum number of comparisons) under R is $H(m)$ ($\lceil H(m) \rceil$).

Lemma 5. If a procedure R is conditionally E-noiseless (conditionally M-noiseless) given the complete pairing on n items then a procedure R' in $\mathscr{P}$ and identical to R after the complete pairing is E-noiseless (M-noiseless) among all possible procedures.

Proof. We prove that the procedure R' is E-noiseless, the parenthetical case can be shown similarly. Since there are $m = 2c(n!) + 1$ permutations consistent with the complete pairing of n items, where $c(i)$ is defined by (1.47); and since R is conditionally E-noiseless, the expected number of comparisons under R after the complete pairing is $H(2c(n!) + 1)$. Furthermore, the procedure R' requires $\pi(n!)$ additional comparisons where $\pi(i)$ is defined by (1.47). Hence

$$A(n|R') = \pi(n!) + H(2c(n!) + 1). \tag{1.53}$$

To prove that R' is an E-noiseless procedure among all possible procedures we need to prove

$$H(n!) = \pi(n!) + H(2c(n!) + 1). \tag{1.54}$$

To show (1.54) we note that by (1.5) and (1.47) with $i = n!$,

$$\begin{aligned} H(n!) &= \lceil \log(n!) \rceil - \frac{2^{\lceil \log(n!) \rceil} - n!}{n!} \\ &= \pi(n!) + \lceil \log(2c(n!) + 1) \rceil - \left(\frac{2^{\lceil \log(2c(n!)+1) \rceil} - (2c(n!)+1)}{2^{\pi(n!)}(2c(n!)+1)} \right) 2^{\pi(n!)} \\ &= \pi(n!) + H(2c(n!) + 1). \end{aligned}$$

□

Sobel has shown in Section 2 of [16] that for $t = 2$ and $n \neq 2^r$ the class $\mathscr{P}_t$ may not be an E-complete class. He has given a procedure R_E that has expectation $6 + \frac{1}{2}$ for $n = 6$, which is smaller than $6 + \frac{2}{3}$, the expectation of an E-optimal procedure in $\mathscr{P}_2$. An E-optimal procedure in $\mathscr{P}_2$ and $n = 6$ can be found by considering all possible continuations after the complete pairing.

CHAPTER II
On the Inductive Class $\mathcal{I}$

2.1. Introduction

Here we consider in detail the problem formulated in Section 1.1 for the special case $t = n - 1$. Apparently, Steinhaus was the first to introduce the problem in [17] with M-optimality criteria, where he heuristically suggested the procedure that we call R_S and explain in the following paragraph. If m items are ordered as $x_1 < x_2 < \ldots < x_m$ the *rank* of x_{m-i+1} is defined to be i; and the median of m items, for even m, is either one of items with rank $\frac{m}{2}$ or $\frac{1}{2}(m+2)$.

For any j with $2 \le j \le n$, we first rank $j-1$ items inductively. Then find the median of $j-1$ items already ranked. Now insert the j^{th} item among the $j-1$ items already ranked by comparing it with the median of the $j-1$ items; if it is larger then compare it with the median of the 'upper group'; if it is smaller then compare it with the median of the 'lower group'. Continue the process until the proper place of the j^{th} item is found. Let $M(n) \equiv M(n|R_S)$. Then

$$(2.1)\quad s_n \equiv M(n) - M(n-1) = \lceil \log(n) \rceil = \lceil H(n) \rceil = 1 + \lfloor \log(n-1) \rfloor,$$

where $M(1) = 0$.

The quantity $s_n = M(n) - M(n-1)$ is in fact the maximum number of comparisons needed to insert one item among $n-1$ items already ranked under the R_S procedure. From (2.1) one can obtain as in [4] the Steinhaus result:

$$(2.2)\quad M(n) = 1 + ns - 2^s,$$

where $s = s_n$ is given by (2.1). In the 1950 edition of [17] Steinhaus conjectures that R_S is an M-optimal procedure, but this remark is removed in the 1960 edition. In [18] he gives a procedure for $n = 5$ that requires at most 7 comparisons to rank 5 items, whereas $M(5) = 8$. Hence R_S is not an M-optimal procedure.

Let $A(n) = A(n|R_S)$. It has been noted by several authors, e.g., in [8], [9], and [11], that

$$(2.3)\quad A(n) - A(n-1) = H(n)$$

where $A(1) = 0$, and hence

$$(2.4)\quad A(n) = \sum_{j=2}^{n} H(j).$$

Here $A(n) - A(n-1)$ is the expected number of comparisons needed for inserting one item among the $n-1$ already ranked under the R_S procedure. The procedure given in [18] is the same as the R_{FJ} procedure in [5] for $n = 5$. It can be shown that $A(5|R_{FJ}) = 6 + \frac{14}{15} < 7 + \frac{1}{15} = A(5)$. Hence the R_S procedure is not E-optimal either.

2.2. The inductive procedure R_I and its expectation

As is mentioned in Section 2.1 the R_S procedure is neither M-optimal nor E-optimal among all possible procedures. Nevertheless, in this section we prove, among other things, that the R_S procedure is both E-optimal and M-optimal in the class $\mathcal{I}$ of inductive procedures.

We now wish to find an inductive procedure R_I, using the technique of dynamic programming, which is E-optimal in $\mathcal{I}$. The method will produce all inductive procedures that are E-optimal in $\mathcal{I}$.

Let $S(n) \equiv A(n|R_I)$ and let

(2.5) $\Delta S(n) = S(n) - S(n-1)$.

We define R_I for $n \geq 2$ by the recursive formula

(2.6) $\Delta S(n) = 1 + \min_{1 \leq y \leq n-1} \left(\frac{y}{n} \Delta S(y) + \frac{n-y}{n} \Delta S(n-y) \right)$

and the boundary conditions

(2.7) $S(0) = S(1) = 0$.

The basic idea in using this method of recursion was introduced by Sobel in [14], [15], and also in a previous paper on group testing referred to as reference 7 in [15]. In fact, let $n-1$ items be ranked inductively. Then the LHS of (2.6) denotes the minimum additional expected number of comparisons needed to insert the n^{th} item under R_I. To get the RHS of (2.6) first we compare the n^{th} item with the one of rank y among the $n-1$ items already ranked. The probability that the n^{th} item being larger in the first comparison is $\frac{y}{n}$, due to the initial randomness of the n items. Given that the n^{th} item is larger than the item of rank y the expected number of comparisons for finding its proper place among the $y-1$ items is $\Delta S(y) = S(y) - S(y-1)$. Similarly, $\frac{n-y}{n}$ is the probability that the n^{th} item is smaller than the one of rank y. Given the latter event, the expected number of comparisons for finding its proper place among $n-y-1$ items is $\Delta S(n-y) = S(n-y) - S(n-y-1)$. The integer l on the RHS of (2.6) represents the 'present' comparison of the n^{th} item and the one of rank y.

The formulas (2.6) and (2.7) define the whole procedure R_I. For $n \geq 2$ at least one y, for $1 \leq y \leq n-1$, is found and the procedure then is defined inductively for all n. Thus the problem of finding an E-optimal procedure in $\mathcal{F}$ is reduced to solving the recursive formula (2.6) with boundary condition (2.7).

If we define for $j \geq 1$

(2.8) $h(j) = j(S(j) - S(j-1))$,

then for $n \geq 2$, (2.6) can be written in the form

(2.9) $h(n) = n + \min_{1 \leq y \leq n-1} (h(y) + h(n-y))$

with the boundary condition

(2.10) $h(1) = 0$.

The recursive relation (2.9) is exactly the same as (2.13) in [14], (4.1) in [15], and (6.3) in [16] where it was investigated extensively by Sobel. Here we list some results of this observation.

(a) For any $n \geq 2$ an integer $y = y(n)$ will yield the minimum in (2.9) if and only if there is no power of 2 strictly between y and $n-y$ (see Lemma 2 in Section 6 of [16]). The set of all such y values has already been denoted by $\Delta_n^{(E)}$ in (1.23).

(b) The set $\Delta_n^{(E)}$ always includes the median, say $\left\lfloor \frac{n}{2} \right\rfloor$, of 1, 2, …, $n-1$.

(c) Using the boundary condition (2.10) we obtain

(2.11) $h(n) = n(1 + \lceil \log(n) \rceil) - 2^{\lceil \log(n) \rceil}$.

Setting $h(n) = n\Delta S(n)$ we find that $\Delta S(n) = H(n)$ given in (1.15).

Hence using (2.7) and (2.4) it follows that

(2.12) $S(n) = \sum_{j=2}^{n} H(j) = A(n)$.

We summarize the results of this section by the theorem below.

Theorem 3. To rank n items any inductive procedure is E-optimal in $\mathcal{I}$ if and only if it compares the n^{th} item with any item whose rank, among $n-1$ items already ranked, belongs to $\Delta_n^{(E)}$. The common minimum expected number of comparisons for all these E-optimal procedures is given by (2.12).

Now we want to use the above technique to find all M-optimal procedures in $\mathcal{I}$. Let R_I' denote any M-optimal inductive procedure in $\mathcal{I}$ and let

(2.13) $\Delta W(n) = M(n|R_I') - M(n-1|R_I')$,

where $W(1|R_I') = 0$. Then for $n \geq 2$, $\Delta W(n)$ satisfies the recursive relation

(2.14) $\Delta W(n) = 1 + \min_{1 \leq y \leq n-1} \max(\Delta W(y), \Delta W(n-y))$

and boundary conditions

(2.15) $W(0) = W(1) = 0$.

Thus the problem of finding the M-optimal inductive procedure R_I' is reduced to solving the recursive formula (2.14) and boundary conditions (2.15).

The formula (2.14) is investigated above in Section 1.3 and is shown there that

(a′) For any $n \geq 2$ an integer $y = y(n)$ will yield the minimum in (2.14) if and only if $y \in \Delta_n^{(M)} = \left[n - 2^{\lceil \log(n) \rceil}, 2^{\lceil \log(n) \rceil} \right]$.

(b′) $\Delta_n^{(M)}$ includes the median, say $\left\lfloor \frac{n}{2} \right\rfloor$, of 1, 2, …, $n-1$.

(c′) Furthermore,

(2.16) $\Delta W(n) = \lceil \log(n) \rceil$

and hence

(2.17) $M(n|R_I') = \sum_{j=2}^{n} \lceil \log(j) \rceil = \sum_{j=2}^{n} \lceil H(j) \rceil = M(n)$,

where $M(n)$ is given by (2.2).

(d′) By Lemma 4 in Section 1.4 we have $\Delta_n^{(E)} \subset \Delta_n^{(M)}$. It follows from (d′) that any procedure that is E-optimal is $\mathcal{I}$ is also M-optimal in $\mathcal{I}$.

We can summarize the results after (2.13) by the theorem below.

Theorem 4. Any inductive procedure for ranking n items is M-optimal in $\mathcal{I}$ if and only if it compares the n^{th} item with any item whose rank among $n-1$ items already ranked, belongs to $\Delta_n^{(M)}$. The common maximum number of comparisons for all these M-optimal procedures is given by (2.17).

From (b), (b′), (d′), and the definition of R_S we have this corollary:

Corollary 1. The Steinhaus procedure R_S is both E-optimal and M-optimal in $\mathcal{I}$.

Remark 1. In Iverson's book, [7], the R_S procedure above is called 'ranking by insertion'. The author says (and we quote from p. 236 of [7]), "For a random distribution ranking by insertion requires fewer comparisons than any other method…." Apart from the ambiguity of 'fewer comparisons' the above statement is inaccurate. By Corollary 1, the procedure R_S has 'optimal' properties only in the restricted class $\mathcal{I}$. In fact, for $n = 5$ the procedure R_{FJ} in [5] or the counter-example in [18] has maximum number of comparisons equal to $7 < M(5|R_S) < 8$, and it also has expected number of comparisons equal to $6+\frac{14}{15} < A(n|R_S) = 7+\frac{1}{15}$. The paper [16] gives several procedures, each of which has smaller expectation and smaller maximum than R_S.

Remark 2. It follows from the definition given by (1.3) that the E-excess of R_S, denoted by $C(n) \equiv C_{n-1}(n|R_S)$ for the problem of ranking n items, i.e., for $t = n-1$, is given by

$$C(n) = A(n) - L(n) \tag{2.18}$$

(for $L(n) = H(n!)$ this coincides with the E-noise of R_S). From the structure of (2.6) and the solution in (2.12) it follows that $A(n) = L(n|\mathcal{I})$ where $L(n|\mathcal{I}) \equiv L(n-1, n|\mathcal{I})$ is given by (1.48), with $t = n-1$ and $e_{n-1} = \mathcal{I}$. Hence (2.18) takes the form

$$C(n) = L(n|\mathcal{I}) - L(n). \tag{2.19}$$

Similarly, from the definition in (1.14a) for $t = n-1$ with $C_{\text{Max}}(n) \equiv D_{n-1}(n|R_S)$, the structure of (2.14), and the solution (2.17), it follows that

$$C_{\text{Max}}(n) = L_{\text{Max}}(n|\mathcal{I}) - L_{\text{Max}}(n), \tag{2.20}$$

where $L_{\text{Max}}(n|\mathcal{I}) = L_{\text{Max}}(n-1, n|\mathcal{I})$ is given by (1.49) with $t = n-1$ and $e_{n-1} = \mathcal{I}$. We can interpret (2.19) and also (2.20) to indicate that the E-excess (M-excess) of the procedure R_S is caused only by the fact that the class $\mathcal{I}$ is not E-complete (M-complete).

2.3. An explicit expression for $S(n)$ and its asymptotic behavior

From (2.12) and (1.5) we have

$$S(n) = \sum_{j=2}^{n} H(j) = \sum_{j=2}^{n} \lceil \log(j) \rceil - \sum_{j=2}^{n} \frac{2^{\lceil \log(j) \rceil} - j}{j}. \tag{2.21}$$

Since the first summation on the far RHS is in fact $M(n)$ derived by Steinhaus and given in (2.2) above, then

$$S(n) = (1 + \lceil \log(n) \rceil)n - 2^{\lceil \log(n) \rceil} - \sum_{j=2}^{n} \frac{2^{\lceil \log(j) \rceil}}{j}. \tag{2.22}$$

To find an asymptotic expression for $S(n)$ we write the summation in the RHS of (2.22) in a different form. For $2^{s_j-1} < j \le 2^{s_j}$ we have $s_j = \lceil \log(j) \rceil$. Let $s \equiv s_n = \lceil \log(n) \rceil$ so that $2^{s-1} < n \le 2^s$. Then

$$\begin{aligned}\sum_{j=2}^{n} \frac{2^{s_j}}{j} &= \sum_{j=2}^{2^{s-1}} \frac{2^{s_j}}{j} + \sum_{j=2^{s-1}+1}^{n} \frac{2^{s_j}}{j} \\ &= \sum_{i=1}^{s-1} \sum_{j=2^{i-1}+1}^{2^i} \frac{2^{s_j}}{j} + 2^s \sum_{j=2^{s-1}+1}^{n} \frac{2^{s_j}}{j} \\ &= \sum_{i=1}^{s-1} 2^i \sum_{j=2^{i-1}+1}^{2^i} \frac{1}{j} + 2^s \sum_{j=2^{s-1}+1}^{n} \frac{1}{j}.\end{aligned}$$

Hence from (2.2) we obtain

$$S(n) = (1+2^s)n - 2^s - \sum_{i=1}^{s-1} 2^i \sum_{j=2^{i-1}+1}^{2^i} \frac{1}{j} - 2^s \sum_{j=2^{s-1}+1}^{n} \frac{1}{j}. \tag{2.23}$$

Now we can use the following result given in page 125 of [3]:

$$\sum_{j=1}^{m} \frac{1}{j} = \ln(m) + \gamma - \int_m^\infty f(x)dx, \tag{2.24}$$

where $\gamma = .577$ is Euler's constant,

$$0 < \int_m^\infty f(x)dx < \frac{1}{8m^2}, \tag{2.25}$$

and for $m = 1$,

$$\int_1^\infty f(x)dx = \gamma - \frac{1}{2} \doteq .0772. \tag{2.26}$$

From (2.24) we obtain

$$\sum_{j=m_1+1}^{m_2} \frac{1}{j} = \ln\left(\frac{m_2}{m_1}\right) + \frac{1}{2}\left(\frac{1}{m_2} - \frac{1}{m_1}\right) + g(m_2) - g(m_1), \tag{2.27}$$

where

$$g(y) = \int_1^y f(x)dx. \tag{2.28}$$

Setting $m_1 = 2^{i-1}$ and $m_2 = 2^i$ in (2.27) we obtain for the double summation in (2.23)

$$\sum_{i=1}^{s-1} 2^i \sum_{j=2^{i-1}+1}^{2^i} \frac{1}{j} = (2^s - 2)\ln(2) - \frac{1}{2}(s-1) + \sum_{i=1}^{s-1} 2^i \left(g(2^i) - g(2^{i-1})\right). \tag{2.29}$$

We note that

$$\begin{aligned}\sum_{i=1}^{s-1} 2^i \left(g(2^i) - g(2^{i-1})\right) &= \sum_{i=1}^{s-1} \left(2^i g(2^i) - 2^{i-1} g(2^{i-1})\right) - \sum_{i=1}^{s-1} 2^{i-1} g(2^{i-1}) \\ &= 2^{s-1} g(2^{s-1}) - \sum_{i=1}^{s-1} 2^{i-1} g(2^{i-1}).\end{aligned}$$

Also, letting $m_1 = 2^{s-1}$ and $m_2 = n$ in (2.27) we obtain

$$\sum_{j=2^{s-1}+1}^{n} \frac{1}{j} = \ln(2) - (s-1)\ln(2) + \frac{1}{2}\left(\frac{1}{n} - \frac{1}{2^{s-1}}\right) + g(n) - g(2^{s-1}). \tag{2.30}$$

Substituting (2.29) and (2.30) into (2.23), we get

$$\begin{aligned} S(n) &= sn - 2^s \ln(n) + s2^s \ln(2) + n - (1 + 2\ln(2))2^s \\ &\quad + \tfrac{1}{2}s - \tfrac{2^{s-1}}{n} + \tfrac{1}{2} + 2\ln(2) + \textstyle\sum_{i=1}^{s} 2^{i-1} g(2^{i-1}) - 2^s g(n). \end{aligned} \tag{2.31}$$

Using (2.26) and (2.25) we get

$$-g(n) \le \tfrac{1}{2} - \gamma + \tfrac{1}{8n^2} \tag{2.32}$$

and

$$g(2^{i-1}) \le \textstyle\int_{2^{i-1}}^{\infty} f(x)dx + g(2^{i-1}) = \gamma - \tfrac{1}{2}. \tag{2.33}$$

Multiplying (2.33) by 2^{i-1}, summing up, and adding the result to 2^r times (2.32) gives

$$-2^s g(n) + \textstyle\sum_{i=1}^{s} 2^{i-1} g(2^{i-1}) \le \tfrac{1}{2} - \gamma + \tfrac{2^s}{8n^2} \le \tfrac{1}{2} - \gamma + \tfrac{1}{4n}.$$

Substituting the last result into (2.31) and noting that $\frac{2^{s-1}}{n} < 1$ gives

$$S(n) = sn - 2^s \ln(n) + s2^s \ln(2) + n - (1 + 2\ln(2))2^s + \tfrac{1}{2}s + \mathcal{O}(1). \tag{2.34}$$

In particular, for $n = 2^s$ we have $s = \frac{\ln(n)}{\ln(2)} = \log(n)$ and (2.34) takes the form

$$S(n) = n\log(n) - (2\ln(2))n + \tfrac{1}{2}\log(n) + \mathcal{O}(1). \tag{2.35}$$

Expressions similar to the RHS of (2.35) but with fewer terms were obtained by Kislitsyn in [8] and [9].

Remark 1. We could also define a procedure for ranking n items as being *asymptotically E-efficient* (*M-efficient*) if its E-excess (M-excess) approaches zero as $n \to \infty$. By this definition we want to show that R_S is not asymptotically E-efficient (M-efficient).

To establish this claim we make use of $M(n|R_{FJ}) \equiv U(n)$ given in (3.28) below. Of course $U(n)$ being the maximum number of comparisons under a given procedure we have

$$L(n) \le U(n). \tag{2.36}$$

Hence

$$C(n) = L(n|\mathcal{F}) - L(n) \ge S(n) - U(n). \tag{2.37}$$

To show that $C(n)$ does not tend to zero as $n \to \infty$, we take the subsequence $n_k = \frac{1}{3}\left(2^{2k+2} - 1\right)$ and prove that $S(n_k) - U(n_k)$ approaches $+\infty$ as $k \to \infty$. Using $U(n_k) = n_k(2k-1) + k + 1$ as in (3.33) and substituting $s = \lceil \log(n_k) \rceil = 2k + 1$ in (2.35), we obtain

$$S(n_k) - U(n_k) = 2^{2k+1}(2k + 2 - 2\ln(2) - \log(n_k))\ln(2) + k + \mathcal{O}(1). \tag{2.38}$$

To finish the proof it suffices to show that

$$2k + 2 - 2\ln(2) - \log(n_k) > 0. \tag{2.39}$$

Since $n_k < \frac{2^{2k+2}}{3}$ then $\log(n_k) < 2k+2-\log(3)$, and since $1.4 \doteq 2\ln(2) < \log(3) \doteq 1.5$ then (2.39) is true, which proves that R_S is not asymptotically E-efficient.

To prove that R_S is not asymptotically E-efficient we note that

$$(2.40)\quad C_{\text{Max}}(n) = L_{\text{Max}}(n|\mathcal{F}) - L_{\text{Max}}(n) \geq M(n) - U(n) \geq S(n) - U(n).$$

Since it was shown that $S(n)-U(n)$ does not approach 0 as $n \to \infty$ then it follows from (2.40) that $C_{\text{Max}}(n) \nrightarrow 0$ as $n \to \infty$.

Remark 2. Because of the simplicity of R_S, it is frequently used to rank n items with computers. Thus it may be of interest to have an upper bound for its E-excess and M-excess. Using the inequalities (1.9) and (1.15), we obtain

$$(2.41)\quad C(n) \leq \sum_{j=2}^{n} H(j) - H(n!)$$

and

$$(2.42)\quad C_{\text{Max}}(n) \leq \sum_{j=2}^{n} \lceil H(j) \rceil - \lceil H(n!) \rceil.$$

To find asymptotic expressions for the RHS's of (2.41) and (2.42), note that by Stirling's formula,

$$(2.43)\quad \lceil H(n!) \rceil = \lceil \log(n!) \rceil = n\log(n) - \frac{1}{\ln(2)} n + \mathcal{O}(\log(n)).$$

Using (1.5) we get

$$(2.44)\quad H(n!) \geq \lceil \log(n!) \rceil - 1 = n\log(n) - \frac{1}{\ln(2)} n + \mathcal{O}(\log(n)).$$

Hence using (2.34), (2.35), (2.44), (2.43), and (2.17) we obtain

$$(2.45)\quad C(n) \leq \begin{cases} \left(\frac{1}{\ln(2)} - 2\ln(2)\right) n + \mathcal{O}(\log(n)) & \text{for } n = 2^s, \\ \left(\frac{1}{\ln(2)} + 1 - \ln(2)\right) n + \mathcal{O}(\log(n)) & \text{for } n \neq 2^s, \end{cases}$$

and

$$(2.46)\quad C_{\text{Max}}(n) \leq \begin{cases} \left(\frac{1}{\ln(2)} - 1\right) n + \mathcal{O}(\log(n)) & \text{for } n = 2^s, \\ \frac{n}{\ln(2)} + \mathcal{O}(\log(n)) & \text{for } n \neq 2^s. \end{cases}$$

2.4. On an inductive procedure $R_S^{(t)}$

As a first step toward finding a procedure for the problem stated in Section 1.1 with general t, we restrict ourselves to a subclass $\mathcal{I}_t'$ of the class of inductive procedures $\mathcal{I}_t$, defined in Section (1.5). $\mathcal{I}_t'$ is the class of procedures that rank the t largest of $n-1$ items inductively and then insert the n^{th} item among the t largest that are already ranked. In $\mathcal{I}_t$ the n^{th} item can be compared with any of $n-t-1$ other items that are not among the t largest, but not in $\mathcal{I}_t'$.

Let $R_S^{(t)}$ be an E-optimal procedure in $\mathcal{I}_t'$ and let $S_t(n) \equiv A_t(n|R_S^{(t)})$ for fixed t with $1 \leq t \leq n-1$. For $n \geq t+1 \geq 2$ the procedure $R_S^{(t)}$ is defined by the recursive relation

$$(2.47)\quad S_t(n) - S_t(n-1) = 1 + \min_{1 \leq y \leq t} \left(\frac{y}{n} \cdot H(y) + \frac{n-y}{n} \left(S_{t-y}(n-y) - S_{t-y}(n-y-1) \right) \right).$$

The boundary conditions are

$$(2.48)\quad \begin{aligned} &S_0(m) = 0 \text{ for all } m, \\ &S_t(m) = S(m) \equiv A(m|R_S) \quad (t \geq m). \end{aligned}$$

In writing $H(x)$ in (2.47) we automatically assumed that the Steinhaus procedure is used when the n^{th} item is larger than the one of rank y among the t largest items already ranked.

Special cases. For $t = 1$, (2.47) and (2.48) reduce to

$$(2.49)\quad S_1(n) - S_1(n-1) = 1 \text{ with } S_1(1) = 0.$$

From (2.49) we get

$$(2.50)\quad S_1(n) = n - 1 = L(1, n).$$

For $t = 2$, (2.47) and (2.48) reduce to

$$S_2(n) - S_2(n-1) = 1 + \min\Big(\tfrac{n-1}{n}\big(S_1(n-1) - S_1(n-2)\big), \tfrac{2}{n}\Big),$$

or in view of (2.49) we get, for $n \geq 3$,

$$(2.51)\quad S_2(n) - S_2(n-1) = 1 + \min\Big(\tfrac{n-1}{n}, \tfrac{2}{n}\Big) = 1 + \tfrac{2}{n}$$

and

$$(2.52)\quad S_2(2) = 1.$$

Since $\frac{2}{n} \leq \frac{n-1}{n}$ for $n \geq 3$ then under the procedure $R_S^{(2)}$ we compare the n^{th} item with the second largest among $n - 1$ items. The expected number of comparisons under $R_S^{(2)}$ is found easily from (2.51) and (2.52) to be, for all $n \geq 2$,

$$(2.53)\quad S_2(n) = n - 1 + 2\sum_{j=3}^{n} \tfrac{1}{j}.$$

The procedure $R_S^{(2)}$ is the same as R_P in pages 8 and 11 of [16].

For $t = n - 1$, (2.47) and (2.48) reduce to (2.6) and (2.7), so that $R_S^{(n-1)} \equiv R_S$.

The study of the recursive relation (2.47), for general t and n, is lengthy enough to be taken up in a separate work.

CHAPTER III
On the Semi-inductive Procedure R_{FJ}

3.1. Introduction

Ford and Johnson in [5] consider the M-optimal problem for ranking n items. Their procedure, which we call R_{FJ}, is in the subclass $\mathcal{J} \equiv \mathcal{J}_{n-1}$, i.e., the class of procedures that do ordinary pairing and use the same procedure on the $\left\lfloor \frac{n}{2} \right\rfloor$ larger items. Let $U(n) = M(n|R_{FJ})$ and $F(n) = A(n|R_{FJ})$. The maximum $U(n)$ is given implicitly in [5] by the recursive relations

$$\begin{aligned} U(2k) &= k + U(k) + \textstyle\sum_{i=2}^{k} T(i), \\ U(2k+1) &= k + U(k) + \textstyle\sum_{i=2}^{k+1} T(i), \end{aligned} \tag{3.1}$$

with boundary conditions

$$U(1) = 0, U(2) = 1,$$

where

$$T(i) = j \text{ for } t_{j-1} < i \leq t_j \tag{3.2}$$

and

$$t_j = \tfrac{1}{3}\left(2^{j+1} + (-1)^j\right) \quad (j = 1, 2, 3, \ldots). \tag{3.3}$$

It was observed that

$$U(n) = \lceil H(n!) \rceil \text{ for } n \leq 11 \text{ and } n = 20, 21. \tag{3.4}$$

Hence R_{FJ} is M-optimal for those values of n. It is conjectured in [5] that R_{FJ} is M-optimal for all n. Two asymptotic expressions for $U(n)$ based on subsequences $n_k = \left\lfloor \frac{2^k}{3} \right\rfloor$ and $n_k = 2^k$ were given to make it possible to do some comparisons with $M(n|R_S) = M(n)$.

In this chapter we do the following: 1) prove that $U(n) \leq M(n)$ for all n; 2) we find a single explicit expression for $U(n)$; 3) we find a lower bound $J_{\text{Max}}(n|\mathcal{J}) \geq \lceil H(n!) \rceil$ for all procedures in the class $\mathcal{J}$; and 4) observe in Table 1 below that $U(n)$ coincides with $L_{\text{Max}}(n|\mathcal{J})$ for many other values of n in addition to those in (3.4). We also consider the E-optimal problem in the class $\mathcal{J}$ and for $n = 7$ give a procedure in $\mathcal{J}$ that has smaller expectation than R_{FJ}. Thus R_{FJ} is not E-optimal in the class $\mathcal{J}$ and hence it is not E-optimal among all procedures.

The reader should note that in page 229 of [1] there is an incorrect description for R_{FJ}, namely in the second step of the procedure the larger items should be ranked according to the R_{FJ} procedure (semi-inductively), and not by the R_S procedure as stated in [1].

3.2. An explicit expression for $U(n)$

From (3.1) we have, for $k \geq 1$,

$$\begin{aligned} U(2k) - U(2k-1) &= 1 + U(k) - U(k-1), \\ U(2k+1) - U(2k) &= T(k+1). \end{aligned} \tag{3.5}$$

Letting

$$V_n = U(n) - U(n-1), \tag{3.6}$$

then (3.5) can be written as

$$\begin{aligned} V_{2k} - V_k &= 1, \\ V_{2k+1} &= T(k+1), \end{aligned} \tag{3.7}$$

with the boundary condition

$$V_1 = 0. \tag{3.8}$$

Let $\pi_n \equiv \pi(n)$ and $c_n \equiv c(n)$ be non-negative integers defined by (1.47), from which we immediately conclude that

$$\begin{aligned} \pi_{2n} &= 1 + \pi_n, \quad & c_{2n} &= c_n, \\ \pi_{2n+1} &= 0, & c_{2n+1} &= n. \end{aligned} \tag{3.9}$$

Using iteration in (3.7) we get

$$\begin{aligned} V_{2k} &= 1 + \pi_k + T(c_k + 1), \\ V_{2k+1} &= T(k+1). \end{aligned} \tag{3.10}$$

In view of (3.9), the relations listed in (3.10) can be written as

$$\begin{aligned} V_{2k} &= \pi_{2k} + T(c_{2k} + 1), \\ V_{2k+1} &= \pi_{2k+1} + T(c_{2k+1} + 1). \end{aligned} \tag{3.11}$$

From (3.11) and (3.8) we conclude that for $n \geq 1$,

$$V_n = \pi_n + T(c_n + 1), \tag{3.12}$$

where we define

$$T(1) = 0. \tag{3.13}$$

Hence the solution to (3.5) is

$$U(n) = \sum_{i=2}^{n} (\pi_i + T(c_i + 1)), \tag{3.14}$$

where $T(k)$ is defined by (3.2) and (3.13).

Now we prove the following lemma which is very useful for comparing $U(n)$ with $M(n)$. We also use it to find a simpler explicit formula for $U(n)$ in the next section.

Lemma 6. The function T defined by (3.2), (3.3), and (3.13) is (in explicit form), for $n \geq 1$,

$$T(n) = \lfloor \log(3n - 2) \rfloor. \tag{3.15}$$

Proof. Consider n and $j = j(n)$ such that $t_{j-1} < n \leq t_j$ where t_j is the integer given by $t_j = \frac{1}{3}\left(2^{j+1} + (-1)^j\right)$. Then

$$2^j + (-1)^{j-1} \leq 3n - 1 < 2^{j+1} + (-1)^j. \tag{3.16}$$

We consider two cases:

(a) $j = 2m$. In this case (3.16) can be written as

(3.17) $2^{2m} - 2 \le 3n - 2 < 2^{2m+1}$.

Since $3m - 2 \equiv -2 \pmod 3$ and $2^{2m} - 2 \equiv -1 \pmod 3$, it follows that we can add 2 to the left side of (3.17) without altering the inequality. Hence

(3.18) $2^{2m} \le 3n - 2 < 2^{2m+1}$,

i.e., $\lfloor \log(3n-2) \rfloor = 2m = j$ and the lemma is proved in this case.

(b) $j = 2m+1$. In this case (3.16) can be written as $2^{2m+1} \le 3n - 1 < 2^{2m+2} - 1$, which implies

(3.19) $2^{2m+1} \le 3n - 2 < 2^{2m+2} - 2 < 2^{2m+2}$.

Hence $\lfloor \log(3n-2) \rfloor = 2m + 1 = j$ and the lemma is proved.

Using Lemma 6, we obtain from (3.12)

(3.20) $V_n = \pi_n + \lfloor \log(3c_n + 1) \rfloor$.

Now using (1.47) and the fact that $\lceil \log(m) \rceil = 1 + \lfloor \log(m-1) \rfloor$, we obtain from (3.19)

$$\begin{aligned} V_n &= \pi_n - 1 + \lfloor \log(6c_n + 2) \rfloor = \pi_n - 2 + \lceil \log(3(2c_n + 1)) \rceil \\ &= -2 + \lceil \log(3n) \rceil. \end{aligned}$$

Hence

(3.21) $V_n = U(n) - U(n-1) = \left\lceil \log\left(\frac{3n}{4}\right) \right\rceil$.

Lemma 7. $U(n) \le M(n)$ for all n. In particular, the inequality is strict if and only if $n \ge 5$.

Proof. From (2.1) and (3.21) we note that $s_n = M(n) - M(n-1) = \lceil \log(n) \rceil \ge \left\lceil \log\left(\frac{3n}{4}\right) \right\rceil = V_n$. Since $U(n) = \sum_{j=2}^{n} V_j$ and $M(n) = \sum_{j=2}^{n} s_j$ it follows that $U(n) \le M(n)$ for all n. To finish the rest of the lemma it suffices to show that $n = 5$ is the first integer such that $\left\lceil \log\left(\frac{3n}{4}\right) \right\rceil < \lceil \log(n) \rceil$. The latter is shown easily by inspection.

3.3. A simpler explicit expression for $U(n)$

Let for every $i \ge 0$

(3.22) $\tau_i = \frac{1}{6}\left(2^{i+2} + (-1)^{i+1} - 3\right)$.

Lemma 8. For V_n in (3.21) and τ_j, $j \ge 1$, in (3.22), we have

(3.23) $V_n = j - 1$ if $\tau_{j-1} < n \le \tau_j$.

Proof. We note that $\tau_j = \frac{1}{2}\left(t_{j+1} - 1\right)$ where t_j is given by (3.3). Thus if $\tau_{j-1} < n \le \tau_j$ then

(3.24) $t_j < 2n + 1 \le t_{j+1}$.

Applying Lemma 6 to (3.24) we get

(3.25) $j+1 = T(2n+1)$,

where $T(n)$ is given by (3.15). Hence

$$\begin{aligned} j+1 &= \lfloor \log(6n+1) \rfloor = -1 + \lceil \log(6n+2) \rceil \\ &= \lceil \log(3n+1) \rceil = \lfloor \log(3n) \rfloor + 1. \end{aligned}$$

Hence

$$j = \lfloor \log(3n) \rfloor.$$

By the same argument as in Lemma 6 it is easily shown that

(3.26) $\lfloor \log(3n) \rfloor - 1 = \left\lceil \log\left(\frac{3n}{4}\right) \right\rceil$.

This completes the proof of the lemma.

Now we proceed to obtain an explicit expression for $U(n)$. Let $\tau_{j-1} < n \le \tau_j$. Then

$$\begin{aligned} U(n) &= \sum_{i=1}^{n} V_i = \sum_{i=1}^{\tau_j} V_i - \sum_{i=n+1}^{\tau_j} V_i \\ &= \sum_{k=1}^{j} \sum_{i=\tau_{k-1}+1}^{\tau_k} V_i - (j-1)(\tau_j - n). \end{aligned} \tag{3.27}$$

For the summation on the RHS we have

$$\sum_{k=1}^{j} \sum_{i=\tau_{k-1}+1}^{\tau_k} V_i = \sum_{k=1}^{j} (\tau_k - \tau_{k-1})(k-1) = \sum_{k=1}^{j} (k\tau_k - (k-1)\tau_k) - \sum_{k=1}^{j} \tau_k.$$

Hence we obtain for (3.27)

$$U(n) = n(j-1) + \tau_j - \sum_{k=1}^{j} \tau_k = \left(n + \tfrac{1}{2}\right)(j-1) - \tfrac{1}{2}\left(\tau_{j-1} - 2 + 2^j\right).$$

Substituting for τ_{j-1} from (3.22) in the last expression we have, for all $n \ge 1$,

(3.28) $U(n) = n(j-1) + \frac{1}{2}j - \frac{2}{3}(2^j - 1) + \frac{1}{6}\left(\frac{1+(-1)^{j+1}}{2}\right)$

where $j = j(n) = \lfloor \log(3n) \rfloor$.

Although the result for $U(n)$ in Equation 6.15 of [16] is slightly different from $U(n)$ in (3.28) above, it is easy to show that the 2 expressions for $U(n)$ are consistent.

From (3.28) we can readily write an asymptotic expression for $U(n)$, i.e.,

(3.29) $U(n) = nj - n\left(1 + \frac{2^{1+j}}{3n}\right) + \frac{1}{2}\log(n) + \mathcal{O}(1)$,

where the coefficient of n in parentheses lies between 4 and 7, and its particular value depends on the subsequence of n chosen. For the subsequence

(3.30) $n_i = \left\lceil \frac{2^{2i+1}}{3} \right\rceil = \frac{2^{2i+1}+1}{3} \quad (i = 1, 2, \ldots)$

we have $j = 2i + 1$ and by (3.28),

(3.31) $U(n) = 2n(i-1) + i + 2.$

For the subsequence

(3.32) $n_i = \left\lfloor \frac{2^{2i+2}}{3} \right\rfloor = \frac{2^{2i+2}-1}{3} \quad (i = 1, 2, \ldots),$

we have $j = 2i + 1$ and by (3.28),

(3.33) $U(n) = n(2i-1) + i + 1.$

3.4. Lower bounds for all procedures in the class $\mathcal{J}$

First we want to find a lower bound $J_{\text{Max}}(n)$ for the maximum number of comparisons needed for ranking n items under any procedure in the class $\mathcal{J}$. For all procedures in $\mathcal{J}$ we do the ordinary pairing and then rank the $\left\lfloor \frac{n}{2} \right\rfloor$ winners (or larger items) by the same procedure. Let the $\left\lfloor \frac{n}{2} \right\rfloor$ winners be denoted by $x_2 \le x_4 \le x_6 \le \ldots$. Denote the corresponding losers by $x_1, x_3, x_5, \ldots$, and insert them in that order. Then x_{2i+1} has $2i + 1$ different places (or spaces) in which it can go and hence we are left with $1 \cdot 3 \cdot 5 \cdot \ldots \cdot (2k-1)$ or $1 \cdot 3 \cdot 5 \cdot \ldots \cdot (2k+1)$ states of nature depending on whether $n = 2k$ or $n = 2k + 1$, respectively. We derive $J_{\text{Max}}(n)$ by the recursive relations

$$(3.34)\quad \begin{aligned} J_{\text{Max}}(2k) &= k + J_{\text{Max}}(k) + \lceil H(3 \cdot 5 \cdot \ldots \cdot (2k-1)) \rceil, \\ J_{\text{Max}}(2k+1) &= k + J_{\text{Max}}(k) + \lceil H(3 \cdot 5 \cdot \ldots \cdot (2k+1)) \rceil, \end{aligned}$$

and the boundary condition $J_{\text{Max}}(1) = 0$.

The values of $J_{\text{Max}}(n)$ can now be calculated with the help of (1.5), and they are given in Table 1 below.

Table 1

n	12	13	14	15	16	17	18	19	20	21	22	23	24	25
$U(n)$	30	34	38	42	46	50	54	58	62	66	71	76	81	86
$J_{\text{Max}}(n)$	30	34	38	41	45	50	54	58	62	66	71	76	81	85
$\lceil H(n!) \rceil$	29	33	37	41	45	49	53	57	62	66	70	75	80	84

(For $n \le 11$, we find that $U(n) = J_{\text{Max}}(n) = \lceil H(n!) \rceil$.)

Since $\lceil H(n!) \rceil$ is a lower bound for any procedure, $J_{\text{Max}}(n)$ is a lower bound for procedures in $\mathcal{J}$ and $L_{\text{Max}}(n|\mathcal{J})$ is the best that can be achieved, we have, for any procedure like R_{FJ} in $\mathcal{J}$,

(3.35) $\lceil H(n!) \rceil \le J_{\text{Max}}(n) \le L_{\text{Max}}(n|\mathcal{J}) \le U(n).$

If $U(n) = J_{\text{Max}}(n)$ for some n then R_{FJ} is M-optimal in $\mathcal{J}$ for that particular value (or those values) of n. Table 1 shows that R_{FJ} is M-optimal in $\mathcal{J}$ for many values of n, including the ones for which R_{FJ} is M-optimal among all procedures. For example, R_{FJ} is M-optimal in $\mathcal{J}$ (but not necessarily among all procedures) for $n = 12, 13, 14, 17$, etc.

At the end of this section an explicit expression for $J_{\text{Max}}(n)$ and the subsequence $n_j = 2^j - 1$ is found using recursive relations in (3.34) and the boundary condition following (3.34).

Now considering the E-optimal problem for the class $\mathcal{J}$, we find a lower bound $J(n)$ for the expected number of comparisons over all procedures in $\mathcal{J}$. Corresponding to (3.34) we now have

$$(3.36)\quad \begin{aligned} J(2k) &= k + J(k) + H(3 \cdot 5 \cdot \ldots \cdot (2k-1)), \\ J(2k+1) &= k + J(k) + H(3 \cdot 5 \cdot \ldots \cdot (2k+1)), \end{aligned}$$

with the boundary condition $J(1) = 0$.

By a similar argument to that used for (3.35) we have

$$(3.37)\quad H(n!) \le J(n) \le L(n|\mathcal{J}) \le F(n),$$

where $F(n) \equiv A(n|R_{FJ})$.

In Table 2 below, a few values of $J(n)$, derived from (3.36), are given. The values of $F(n)$ were calculated using the definition of R_{FJ} in [5] and are the same values that appear in [16].

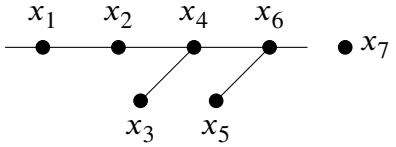
Table 2

n	2	3	4	5	6	7	8	9
$F(n)$	1	$2\frac{2}{3}$	$4\frac{2}{3}$	$6\frac{14}{15}$	$9\frac{27}{45}$	$12\frac{144}{315}$	$15\frac{144}{315}$	$18\frac{1656}{2835}$
$J(n)$	1	$2\frac{2}{3}$	$4\frac{2}{3}$	$6\frac{14}{15}$	$9\frac{27}{45}$	$12\frac{141}{315}$	$15\frac{141}{315}$	$18\frac{1653}{2835}$
$H(n!)$	1	$2\frac{2}{3}$	$4\frac{2}{3}$	$6\frac{14}{15}$	$9\frac{26}{45}$	$12\frac{118}{315}$	$15\frac{118}{315}$	$18\frac{1574}{2835}$

(Values of $F(n)$ for $n = 10, 11, \ldots, 16$ are given in Table 3 later on.)

If $F(n) = J(n)$ for some value(s) of n then R_{FJ} is E-optimal in the class $\mathcal{J}$ for those particular value(s). In Section 1.2, we pointed out that R_{FJ} is not E-optimal among all procedures since we had a counter-example for $n = 6$. In fact, the table in Section 5 of [16] shows that there are several other procedures with smaller expectation than $F(n)$ for $n \le 10$. We now show, in addition, that R_{FJ} is not E-optimal in the class $\mathcal{J}$. The tree of Fig. 4 represents a procedure of $\mathcal{J}$ that is E-optimal in $\mathcal{J}$ for $n = 7$; it has expectation $12 + \frac{141}{315} < F(7) = 12 + \frac{144}{315}$.

For $n = 7$ all the procedures in $\mathcal{J}$ start with ordinary pairing and ranking of the larger items, which requires an average of $5 + \frac{2}{3}$ comparisons and yields the following diagram:



A continuation which leads to a procedure that is in $\mathcal{J}$ is given below in Fig. 4.

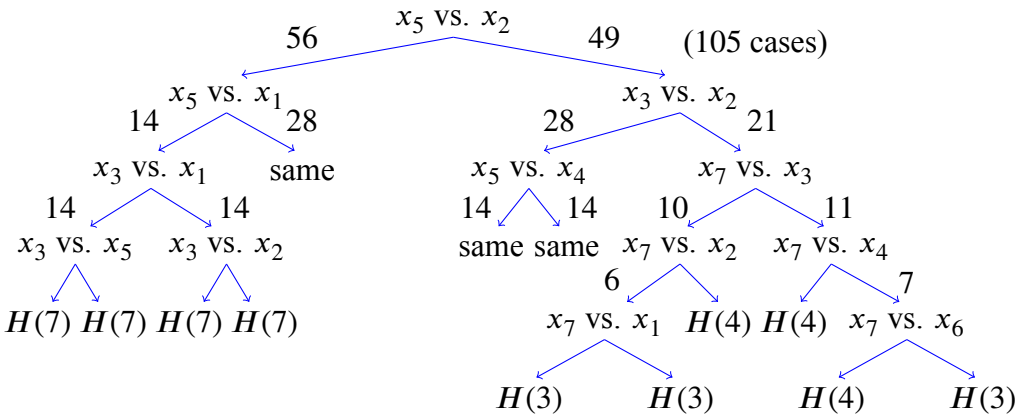


Fig. 4. A continuation that is E-optimal in $\mathcal{J}$.

Now we want to find explicit expressions for $J_{\text{Max}}(n)$ and $J(n)$ using the recursive relations given by (3.34) and (3.36) respectively. It will be clear that (3.34) and (3.36) are special cases of the recursive formula (3.39) below.

Define δ_x by

(3.38) δ_x = the largest odd integer not smaller than n.

Consider for any function f with $f(1) = 0$ the recursive relation

$$I(n) = \left\lfloor \frac{n}{2} \right\rfloor + I\left(\left\lfloor \frac{n}{2} \right\rfloor \right) + f(\delta_n) \tag{3.39}$$

and boundary condition $I(1) = 0$. We note that if $f(\delta_n) = \lceil H(3 \cdot 5 \cdot \ldots \cdot \delta_n) \rceil$ then (3.39) reduces to (3.34), and if $f(\delta_n) = H(3 \cdot 5 \cdot \ldots \cdot \delta_n)$ then (3.39) reduces to (3.36). Hence $J_{\text{Max}}(n)$ and $J(n)$ can be found from $I(n)$.

To solve (3.39) for $I(n)$ we need to show that

$$\underbrace{\left\lfloor \ldots \tfrac{1}{2} \left\lfloor \tfrac{1}{2} \left\lfloor}_{j \text{ brackets}} \tfrac{1}{2} n \right\rfloor \right\rfloor \ldots \right\rfloor = \left\lfloor \frac{n}{2^j} \right\rfloor. \tag{3.40}$$

We show (3.40) using induction on j. For $j = 1$, (3.40) is trivial. It remains to show that

$$\left\lfloor \tfrac{1}{2} \left\lfloor \frac{n}{2^{j-1}} \right\rfloor \right\rfloor = \left\lfloor \frac{n}{2^j} \right\rfloor. \tag{3.41}$$

Case 1. If $n < 2^j$, i.e., $j > \lfloor \log(n) \rfloor$, then the RHS of (3.41) is equal to zero. Since $0 \le \frac{n}{2^{j-1}} < 2$ then $0 \le \left\lfloor \frac{n}{2^{j-1}} \right\rfloor < 2$ and hence

$$0 \le \tfrac{1}{2} \left\lfloor \frac{n}{2^{j-1}} \right\rfloor < 1, \tag{3.42}$$

which shows that the LHS of (3.41) is also zero in this case.

Case 2. $n \ge 2^j$, i.e., $j \le \lfloor \log(n) \rfloor$. Let non-negative integers β and ϵ be defined such that

$$n = \beta 2^j + \epsilon \quad (0 \le \epsilon < 2^j). \tag{3.43}$$

Then for the RHS of (3.41) we have $\left\lfloor \frac{n}{2^j} \right\rfloor = \left\lfloor \frac{\beta 2^j + \epsilon}{2^j} \right\rfloor = \beta$. To find the LHS of (3.42) we note from (3.43) that

(3.44) $2\beta \le \frac{n}{2^{j-1}} < 2(\beta + 1)$.

Hence

(3.45) $\beta \le \frac{1}{2}\left\lfloor \frac{n}{2^{j-1}} \right\rfloor < \beta + 1$,

which implies $\beta = \left\lfloor \frac{1}{2}\left\lfloor \frac{n}{2^{j-1}} \right\rfloor \right\rfloor$. Thus (3.41) is true.

Using (3.40) and iteration in (3.39) we obtain

(3.46) $I(n) = \sum_{j=1}^{\lfloor \log(n) \rfloor} \left\lfloor \frac{n}{2^j} \right\rfloor + \sum_{j=0}^{\lfloor \log(n) \rfloor - 1} f\left(\delta_{\lfloor n/2^j \rfloor}\right)$.

Using (1.46) and the fact that $\delta_x = \delta_{\lfloor x \rfloor}$ we have

(3.47) $I(n) = n - \alpha + \sum_{j=0}^{\lfloor \log(n) \rfloor - 1} f\left(\delta_{n/2^j}\right)$,

where α is the number of 1's in the binary expansion of n.

Letting $f\left(\delta_{n/2^j}\right) = \lceil H(3 \cdot 5 \cdot \ldots \cdot \delta_{n/2^j}) \rceil$ in (3.47) we obtain

(3.48) $J_{\text{Max}}(n) = n - \alpha + \sum_{j=0}^{\lfloor \log(n) \rfloor - 1} \lceil H(3 \cdot 5 \cdot \ldots \cdot \delta_{n/2^j}) \rceil$,

and, letting $f\left(\delta_{n/2^j}\right) = H(3 \cdot 5 \cdot \ldots \cdot \delta_{n/2^j})$ in (3.47), we obtain

(3.49) $J(n) = n - \alpha + \sum_{j=0}^{\lfloor \log(n) \rfloor - 1} H(3 \cdot 5 \cdot \ldots \cdot \delta_{n/2^j})$.

In particular, for the subsequence $n_k = 2^k$ we have $\delta_{n/2^j} = 2^{k-j} - 1$ and $\alpha = 1$, and hence

(3.50) $J_{\text{Max}}(n_k) = n_k - 1 + \sum_{j=0}^{k-1} \lceil H(3 \cdot 5 \cdot \ldots \cdot (2^{k-j} - 1)) \rceil$

and

(3.51) $J(n_k) = n_k - 1 + \sum_{j=0}^{k-1} H(3 \cdot 5 \cdot \ldots \cdot (2^{k-j} - 1))$.

Finally, we note that

(3.52) $J_{\text{Max}}(n_k - 1) = J_{\text{Max}}(n_k) - k - 1$

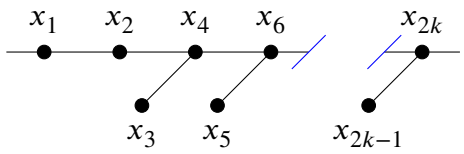
and

(3.53) $J(n_k - 1) = J(n_k) - k - 1$.

3.5. Recursive relations for $F(n) \equiv A(n|R_{FJ})$

After ordinary pairing and semi-induction for $\left\lfloor \frac{n}{2} \right\rfloor$ larger items we are left with one of two configurations, (a) or (b) in Fig. 5, depending upon whether $n = 2k$ or $n = 2k + 1$ respectively.

(a)



(b)

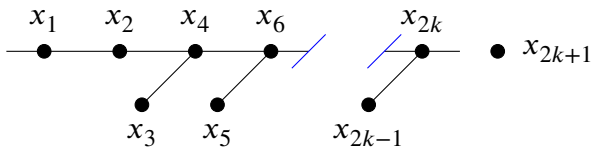


Fig. 5.

Let the order of insertion of the items with indices 3, 5, …, $2\left\lfloor\frac{n-1}{2}\right\rfloor + 1$, according to the R_{FJ} procedure given in [5], be 1, 2, 3, …, $\left\lfloor\frac{n-1}{2}\right\rfloor$, respectively. For convenience, we denote the item to be inserted first by y_1, the next one to be inserted by y_2, etc., and the last to be inserted by $y_{\lfloor(n-1)/2\rfloor}$. For example, if $n = 15$, then the items to be inserted are x_3, x_5, …, x_{15} and the order of insertion under R_{FJ} is x_5, x_3, x_9, x_7, x_{15}, x_{13}, x_{11}; hence $y_1 = x_5$, $y_2 = x_3$, …, $y_7 = x_{11}$.

Let $e_n(\alpha)$ be the expected number of comparisons needed for the insertion of y_α given that the items y_1, y_2, …, $y_{\alpha-1}$ are inserted. Then for $n \geq 2$,

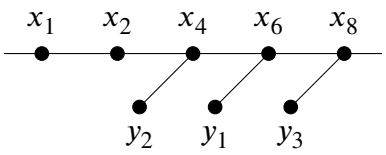
$$F(n) = \left\lfloor\frac{n}{2}\right\rfloor + F\left(\left\lfloor\frac{n}{2}\right\rfloor\right) + \sum_{\alpha=1}^{\lfloor(n-1)/2\rfloor} e_n(\alpha) \tag{3.54}$$

and $F(1) = 0$.

The values of $e_n(\alpha)$ for small α are easy to compute, but for large n (say $n > 16$) and large α (say $\alpha > 7$) the calculations are tedious. For example, we have for $n = 8$

$$e_8(1) = H(4), e_8(2) = \tfrac{1}{5}H(3) + \tfrac{4}{5}H(4), \text{ and } e_8(3) = H(7),$$

where $H(n)$ is given by (1.5). To see this we consider the configuration



It is clear that $e_8(1) = H(4)$, which is the expected number of comparisons needed for insertion of y_1 into the chain $x_1 < x_2 < x_4$ under the R_S procedure. If $x_4 < y_1$ (an event with probability $\frac{3\cdot7}{3\cdot5\cdot7} = \frac{1}{5}$) then $H(3)$ is the expected number of comparisons for insertion of y_2 in the chain $x_1 < x_2$. If $y_1 < x_4$ (an event with probability $\frac{3\cdot4\cdot7}{3\cdot5\cdot7} = \frac{4}{5}$) then $H(4)$ comparisons is the average needed for insertion of y_2 in the chain of 3 items, x_1, x_2, and y_1, already ranked. Finally, we need an average of $H(7)$ for inserting y_3 in the chain of 6 items, x_1, x_2, x_4, x_6, y_1, y_2, already ranked. In a similar manner for $n = 9$ we find that

$$\begin{aligned} &e_9(1) = e_8(1) = H(3), e_9(2) = e_8(2), e_9(3) = H(8), \text{ and} \\ &e_9(4) = \tfrac{1}{9}H(7) + \tfrac{8}{9}H(8). \end{aligned} \tag{3.55}$$

Define an integral valued function $j = j(n)$ for $n \geq 2$ by

(3.56) $2t_j \leq n < 2t_{j+1}$,

where t_j is given by (3.3). It is not difficult to see that for n even and all $\alpha \geq 1$

(3.57) $e_n(\alpha) = e_{n-1}(\alpha)$;

and for $1 \leq \alpha \leq t_j - 1$ and all n,

(3.58) $e_n(\alpha) = e_{n+1}(\alpha)$.

Using (3.57) and (3.58) it follows from (3.54) that for $k \geq 1$,

$$\begin{aligned} &F(2k+1) - F(2k) = \textstyle\sum_{\alpha=t_j}^{k} e_{2k+1}(\alpha) - \sum_{\alpha=t_j}^{k-1} e_{2k}(\alpha), \\ &F(2k) - F(2k-1) = 1 + F(k) - F(k-1), \end{aligned} \tag{3.59}$$

where for convenience we set for $t_j > k - 1$

(3.59a) $\sum_{\alpha=t_j}^{k-1} e_{2k}(\alpha) = 0$.

Here the boundary conditions for F are

(3.60) $F(1) = F(0) = 0$.

The values of $F(n)$ for $2 \leq n \leq 9$ are given in Table 2, Section 3.4. The following values of $e_n(\alpha)$ are needed to extend Table 2 for $10 \leq n \leq 16$. Since $2t_3 = 10$ and $2t_4 = 22$, then for all $n = 10$, 11, …, 16 we have $j = 3$ and $t_j = 5$.

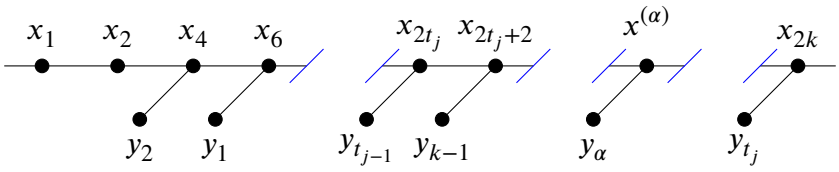
$$\begin{aligned} &e_{11}(5) = H(11) = e_{12}(5), \\ &e_{13}(5) = H(12) = e_{14}(5), e_{13}(6) = \tfrac{1}{13}H(11) + \tfrac{12}{13}H(12) = e_{14}(6), \\ &e_{15}(5) = H(13), e_{15}(6) = \tfrac{1}{15}H(12) + \tfrac{14}{15}H(13), \\ &e_{15}(7) = \tfrac{3}{13\cdot 15}H(11) + \left(\tfrac{12\cdot 2}{13\cdot 15} + \tfrac{12}{13\cdot 15}\right)H(12) + \tfrac{12\cdot 13}{13\cdot 15}H(13). \end{aligned} \tag{3.61}$$

Table 3

Expected number of comparisons under the R_{FJ} procedure (continuation of Table 2 above)

n	10	11	12	13	14	15	16
$F(n)$	$21\frac{268}{315}$	$25\frac{1373}{3465}$	$29\frac{218}{3465}$	$32\frac{37904}{45045}$	$36\frac{31469}{45045}$	$40\frac{29418}{45045}$	$44\frac{29418}{45045}$

Now we want to obtain more specific results about $e_n(\alpha)$. For $n = 2k$ (with $2t_j < n < 2t_{j+1}$) the diagram, after the 1st and 2nd step of the R_{FJ} procedure, is



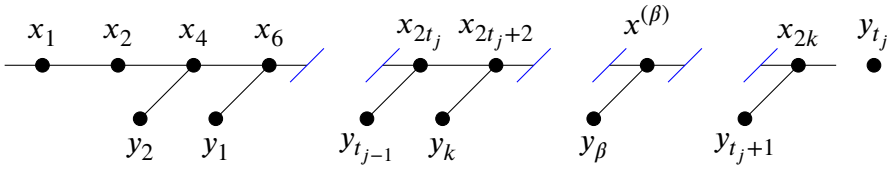
For every α such that $t_j < \alpha \le k-1$ let $x^{(\alpha)}$ be the item directly larger than y_α, and let $N(x^{(\alpha)})$ denote the number of items from the set $\{y_{t_j}, y_{t_j+1}, \dots, y_{\alpha-1}\}$ that are smaller than $x^{(\alpha)}$. Define

$$p_\lambda(\alpha, 2k) = \text{Prob}\{N(x^{(\alpha)}) = \lambda | R_{FJ}\} \tag{3.62}$$

for $\lambda = 0, 1, 2, \dots, \alpha - t_j$. For convenience, let $p_0(t_j, 2k) = 1$. Then $\sum_{\lambda=0}^{\alpha-t_j} p_\lambda(\alpha, 2k) = 1$ for all α with $t_j \le \alpha \le k-1$. Hence

$$e_{2k}(\alpha) = \sum_{\lambda=0}^{\alpha-t_j} p_\lambda(\alpha, 2k) H(2t_j + k - \alpha + \lambda). \tag{3.63}$$

Similarly, for $n = 2k+1$ (with $2t_j < n < 2t_{j+1}$) the diagram is



For every $t_j < \beta \le k$ let $x^{(\beta)}$ be the item directly larger than y_β, and let $N(x^{(\beta)})$ denote the number of items from the set $\{y_{t_j}, y_{t_j+1}, \dots, y_{\beta-1}\}$ that are smaller than $x^{(\beta)}$. Define

$$p_\nu(\beta, 2k+1) = \text{Prob}\{N(x^{(\beta)}) = \nu | R_{FJ}\} \tag{3.64}$$

for $\nu = 0, 1, \dots, \beta - t_j$. For convenience, let $p_0(t_j, 2k+1) = 1$. Then $\sum_{\nu=0}^{\beta-t_j} p_\nu(\beta, 2k+1) = 1$ for all β with $t_j \le \beta \le k$. Hence

$$e_{2k+1}(\beta) = \sum_{\nu=0}^{\beta-t_j} p_\nu(\beta, 2k+1) H(2t_j + k - \beta + \nu + 1). \tag{3.65}$$

In (3.61) a few values of $e_{2k}(\alpha)$ and $e_{2k+1}(\beta)$ are given. Substituting (3.63) and (3.65) into (3.59) and letting $\alpha - \lambda = i$ and $\beta - \nu = m$, we obtain

$$\begin{aligned} F(2k+1) - F(2k) &= \sum_{m=t_j}^{k} \Big(\sum_{\beta=m}^{k} p_{\beta-m}(\beta, 2k+1) \Big) H(2t_j + k + 1 - m) \\ &\quad - \sum_{i=t_j}^{k-1} \Big(\sum_{\alpha=i}^{k-1} p_{\alpha-i}(\alpha, 2k) \Big) H(2t_j + k - i), \\ F(2k) - F(2k-1) &= 1 + F(k) - F(k-1). \end{aligned} \tag{3.66}$$

Since $F(2k+1) - F(2k)$ is not a simple expression it is useful to obtain bounds for it. Since $H(x)$ is an increasing function of x, it follows from (3.63) and (3.65) that

$$H(2t_j + h - \alpha) \le e_{2k}(\alpha) \le H(t_j + k) \tag{3.67}$$

and

$$H(2t_j + k - \beta + 1) \le e_{2k+1}(\beta) \le H(t_j + k + 1). \tag{3.68}$$

Using (3.67) and (3.68) we obtain from (3.59)

$$\begin{aligned} &\textstyle\sum_{\beta=t_j}^{k} H(2t_j + k - \beta + 1) - (k - t_j)H(t_j + k) \le F(2k+1) - F(2k) \\ &\le (k + 1 - t_j)H(t_j + k + 1) - \textstyle\sum_{\alpha=t_j}^{k-1} H(2t_j + k - \alpha). \end{aligned} \tag{3.69}$$

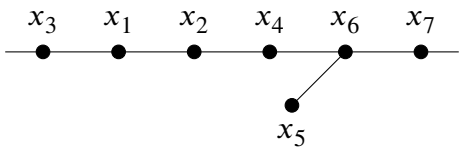
It is interesting to note that from the definition of the R_{FJ} procedure,

$$F(n+1) - F(n) = H(n+1), \text{ provided that } n = 2t_j. \tag{3.70}$$

The relation (3.70) is also implied by (3.69); because for $n = 2t_j$, i.e., $k = t_j$, both the lower and upper sides of (3.69) are equal to $H(2t_j + 1)$ and hence (3.70) follows. Hence the bounds in (3.69) can not be improved.

APPENDIX
Remarks on Trees Considered in the Text

For all of our trees (in particular, for the tree in Fig. 2) the symbol $H(m)$ at the end of a branch at any level means that there is a simple continuation of that branch which requires $H(m)$ additional comparisons on average, where $H(m)$ is given by (1.5). In this simple continuation, we have to rank 1 item among $m-1$ items already ranked, and the procedure used for this purpose is R_S, described in Section 2.1. In Chapter 2 it is proved that R_S is an E-optimal (M-optimal) continuation and it requires $H(m)$ comparisons on average (maximum of $\lceil H(m) \rceil$ comparisons). For example, the continuation $H(5)$ for the branch marked $(*)$ in Fig. 2 is given in Fig. A.1 below. For this $H(5)$-situation we have the diagram



which indicates that $x_3 < x_1 < x_2 < x_4 < x_6 < x_7$ and $x_5 < x_6$; and the continuation:

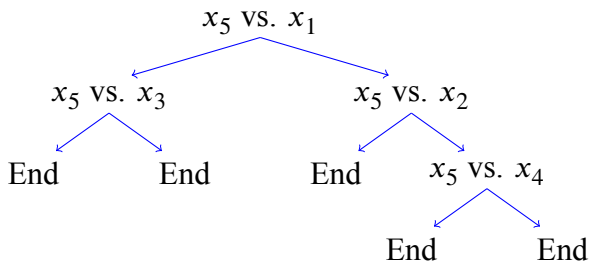


Fig. A.1. Continuation of the branch marked $(*)$ in Fig. 2.

The parenthetical '(315 cases)' at the root of the tree in Fig. 2 is the number of permutations of 7 items x_i $(i = 1, 2, \ldots, 7)$ consistent with the conditions

$$x_1 < x_2 < x_4, x_3 < x_4, x_5 < x_6,$$

which can also be represented by Fig. A.2.

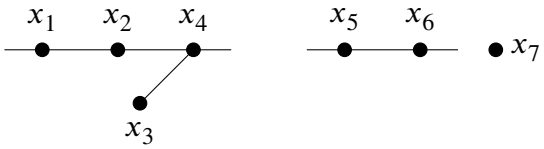


Fig. A.2.

Fig. A.2 indicates the results of complete pairing of 7 items. Other numbers at different stages have a similar meaning, i.e., they represent the number of permutations of 7 items, subject to the inequalities known at that stage.

The word 'same', written at the end of some branches, means that the continuation of that branch is essentially (except for the renaming of some items) the same as the continuation of the branch on the same level and the same structure (diagram), and consequently with the same number of cases.

The numbers in a circle, which appear only when they are non-zero, indicates the number of powers of 2 between two numbers accompanying the arrows. For example, the encircled number $\textcircled{+1}$ between the arrows accompanying the partition $(84, 63)$ indicates that there is exactly one power of 2 (namely $2^6 = 64$) between 84 and 63. The partition $(84, 63)$ is the result of the x_3 vs. x_2 comparison, which divides the 147 states of nature into 2 sets depending on whether $x_3 < x_2$ or $x_3 > x_2$.

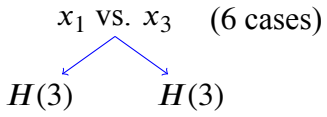
To compute the expected length of the tree in Fig. 2 we use a simple formula due to Sobel (see (6.13) in [16]), putting in 3 for the number of noise units and using (1.5). This formula gives for any tree T the expected length $E(L|T)$:

$$\text{(A.3)}\quad E(L|T) = H(m) + \frac{U}{m} = \left(8 + \frac{118}{315}\right) + \frac{3}{315} = 8 + \frac{121}{315},$$

where m is the number of cases at the root of the tree. Finally, since we made 4 comparisons prior to the tree in Fig. 2, the expected number of comparisons for ranking 7 items according to the procedure R_1 is

$$\text{(A.4)}\quad A(7|R_1) = 12 + \frac{121}{315}.$$

The introduction of $H(m)$ into the situation greatly simplifies a tree. For example, with this notation the tree in Fig. 1 takes the simple form



and by (1.5) and (A.3) with $U = 0$ and $m = 6$, the expected length is $H(6) = 2 + \frac{2}{3}$.

REFERENCES

[1] R. G. Busacker and T. L. Saaty. *Finite graphs and networks : an introduction with applications.* International series in pure and applied mathematics. McGraw-Hill, 1965.

[2] Y. Cesari. *Questionnaire, codage et tris.* Problèmes de traitement de l'information 4. Centre National de la Recherche Scientifique, Institut Blaise Pascal, [Paris], 1968.

[3] H. Cramér. *Mathematical methods of statistics.* Princeton University Press, 1946.

[4] H. A. David. *The Method of Paired Comparisons.* Hafner Pub. Co., 1963.

[5] L. R. Ford Jr. and S. M. Johnson. A tournament problem. *The American Mathematical Monthly*, 66(5):387–389, 1959.

[6] D. A. Huffman. A method for the construction of minimum-redundancy codes. *Proceedings of the IRE*, 40(9):1098–1101, 1952.

[7] K. E. Iverson. A programming language. In *Proceedings of the May 1-3, 1962, Spring Joint Computer Conference*, AIEE-IRE '62 (Spring), page 345–351, New York, NY, USA, 1962. Association for Computing Machinery.

[8] S. S. Kislitsyn. К оценке наименьшего среднего числа попарных сравнений, необходимых для полного упорядочения предметов различного веса // Вестн. ЛГУ. Сер. математики, механики и астрономии. 1962. 1(1):162–163.

[9] S. S. Kislitsyn. Уточнение оценки наименьшего среднего числа сравнений, необходимых для полного упорядочения конечной совокупности // Вестн. Ленингр. ун–та. Сер. математики, механики и астрономии. 1963. 19(4):143–145.

[9a] J. W. Moon. *Topics on tournaments.* Athena series : selected topics in mathematics. Holt, Rinehart, 1968.

[10] C. S. Ogilvy. *Tomorrow's math : unsolved problems for the amateur.* Oxford University Press, 1962.

[11] C. F. Picard. *Théorie des questionnaires.* Les Grands problèmes des sciences, no. 20. Gauthier–Villars, Paris, 1965.

[12] M. Sandelius. On an optimal search procedure. *The American Mathematical Monthly,* 68(2): 133–134, 1961.

[13] J. Schreier. O systemach eliminacji w turniejach. Mathesis Polska, 7(9–10):154–160, 1932.

[14] R. Busacker and T. Saaty, Finite Graphs and Networks: An Introduction with Applications. New York: McGraw-Hill, 1965.

[15] M. Sobel. *Optimal Group Testing.* 1966.

[16] M. Sobel. *On the Ordering of the t = 2 Best of n Items Using Binary Comparisons.* 1968.

[17] H. Steinhaus. *Mathematical snapshots.* Oxford University Press, 1950.

[18] H. Steinhaus. *One Hundred Problems in Elementary Mathematics.* Pergamon Press, Oxford, 1963.

[19] H. Steinhaus. Some remarks about tournaments. In *Golden Jubilee Commemoration Volume (1958-1959), Part II,* page 323–327, Calcutta Mathematical Society, Calcutta, 1959.